

Oric Owner



DECEMBER/JANUARY 1985 · ISSUE 10

FIRST LOOK AT THE ORIC IQ164

INS & OUTS OF THE ORIC MODEM

CHRISTMAS CAROLS TO TYPE IN



Totally devoted to you!

Oric Owner is the official magazine devoted to the Oric 1 and Atmos home computers.

It's crammed full of in-depth information, advance news on the latest add-ons, superb programs and interviews with the engineers who designed it.

The first issue is absolutely free when you buy your Oric, so why not keep ahead of the latest developments and subscribe to further issues. A year's subscription of 6 issues is now only £7.50 so post the coupon today.

Can you imagine life without it?



ORIC OWNER SUBSCRIPTION FORM

Please send me the next 6 issues

I enclose a cheque for (cheques payable to Tansoft Ltd)

Name.....

Address.....

..... Postcode.....

Back issues are available for £1.20 each. If you missed your first free issue contact your dealer or Oric Products International who will supply you with one.

TANSOFT

Tansoft Ltd, Units 1 & 2
Cambridge Techno Park Newmarket Road, Cambridge CB5 8PB

Contents

DECEMBER/JANUARY – ISSUE 10

COMMENT Page 4

Would you like to see a machine code problem page?

NEWS Page 5

Includes a new machine from Oric and Oric joystick and interface details.

CLUB SPOT Page 7

A chance for your club to be printed or maybe you would like to contact other clubs in Europe as well as the U.K.

SOFTWARE SCAN Page 9

Plenty of reviews from Severn, Softbacks, Knight Software, and a first look at titles from No Mans Land.



ORIC MODEM Page 54

If you are thinking of buying an Oric modem this article is a must.

ORIC CLUB Page 56

Plenty of offers on tapes, books and Oric peripherals for you to choose from.

ORIC QUICKIES Page 58

These are for those who get tired of typing in the longer programs.

HINTS & TIPS Pages 60, 63

Several small programs and useful ideas to help you with your Oric.

THE NEW ORIC Page 61

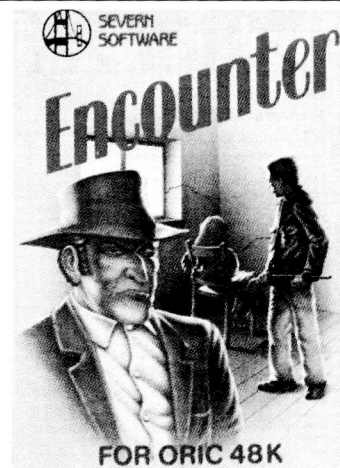
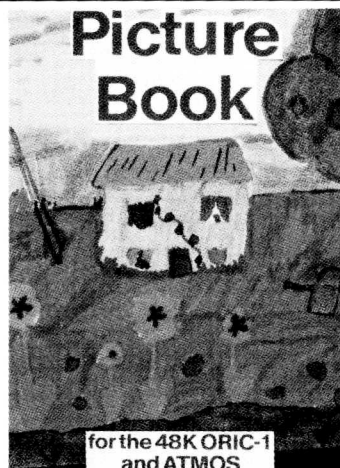
All the facts and details of the new Oric IQ164

LETTERS Page 64

A variety of letters from Oric owners.

ORIC WINS £1,000.00 Page 65

An interesting article explaining how this person wrote a program with his Oric and won 1st prize in a competition.



SOFTWARE LISTINGS Page 14

Eight programs that should cater for everyone, including an appropriate one for Christmas

CHRISTMAS CAROLS – A. Thorn

FOOTBALL POOLS – E. Hollister

ETCH-A-SKETCH – P. Cooper

PITFALL – P. Turner

REVENGE OF THE BEANS – D. Peat

MIDNIGHT FEAST – D. Reid

CRIBBAGE – L. G. Rooke

ATMOS BASIC UTILITIES – D. Sinfield

EXTENSION ROM BOARD Page 51

Add up to 128K of extra ROM to your Oric.

BOOK REVIEWS Page 53

A look at Oric and non-Oric titles that you find useful.



BITZ & PIECES Page 67

Classified Adverts, a puzzle and cartoon.

ADVERTISING

QUARK DATA

Back Cover

TANSOFT

Pages 12 and 13

SOFTBACKS

Page 13

ADDER PUBLISHING

Page 13

Cover illustration by Carolyn Groeneveld

Editor's Comment

Oric Owner

Editor: Carolyn Groeneveld
Graphics: Carolyn Groeneveld
Printers: Burlington Press
(Cambridge) Ltd, Foxton, Cambridge



Well! Christmas has been and gone and I must say I enjoyed the break. I've been desperately trying to get all the news, latest software and plenty of good programs together for you to enjoy in the New Year.

I've managed to get details on the new Oric Joystick and Interface as well as reviews on the Oric Modem and most importantly the new Oric IQ164. I think you'll agree, that's more than we usually have in one issue.

Quite a few people have been asking for the names and addresses of the software houses to go with the long list of software available for the Oric-1/Atmos, printed in the last issue. So, if you were one of them, your problems are solved as they're printed in this issue.

You should all receive your Oric Club membership card with this issue and you'll notice the new offers pages, which I'm sure you won't want to miss out on. If you know of any accessories that you would like to see on offer, please let me know – I'm always open to new ideas. One important thing I'd

like you to do is let me know if you **DO NOT** want your name and address to be given to other club members, as I do not want to upset any of you! I'll wait until the next issue before sending out any names and addresses to club members, so you've got plenty of time.

The software for you to type in includes "Christmas Carols", which I thought was rather appropriate. Apart from that I've tried to include something for everyone. We have had hundreds of programs submitted to us and I must thank you all for that. The only two types of program we don't receive much of are Forth and Mathematical programs. So if you have written any Forth or Mathematical programs that you would like us to publish, please send them in. You'll find the procedure for program submission on the Index page.

I'd also like to start a problem page. If you have any questions or problems specifically about programming and Machine Code, write to Oric Owner and if we get enough we'll print the replies on this problem page. This is something that we have not tried before but I feel sure that it will be worthwhile with your help.

All that's left for me to say now is that I hope you all enjoy the New Year.

Happy 1985.

Carolyn Groeneveld

Oric Owner is published at bi-monthly intervals by Tansoft Ltd.
Units 1 & 2,
Cambridge Techno Park,
Newmarket Road,
Cambridge
Tel: 02205 2261
Reg. No. 1632070
ISSN 0265-8488
Subscriptions: £7.50 for six issues (UK only).
Overseas rates on request
Backnumbers: £1.20 (7, 8 & 9 only)
© Tansoft Ltd.

NEW SOFTWARE HOUSE LAUNCHED

Well known in the Oric Software field, Paul Kaufman, Cathie Burrell and Geoff Phillips have joined together to form a new software house – ORPHEUS.

The company based in Hatley St. George, 15 miles from Cambridge, will be launching initially with three games titles for the Oric 1/Atmos to be rapidly followed up by titles for the ZX Spectrum, Commodore 64 and MSX machines. A business software division of the company will be set up to produce professional products for the Sirius, Apricot and other similar machines.

Paul Kaufman, Managing Director of the Company was formerly a Director of Tansoft Ltd and Editor of the Oric Owner magazine.

TANSOFT'S NEW TITLE

Insect Insanity is the latest game from John Marshall, who brought you that other game for budding psychopaths – Rat Splat.

The aim of Insect Insanity is to protect the jam in the centre of the screen from marauding insects. You do this by stamping on them with a large boot, which you control to the continuous accompaniment of "Flight of the Bumble Bee".

When the game starts you only have creeping slime and the occasional cocoon to stamp on but as the game progresses, the number and type of creatures increases and includes such things as the giant rabid boot cruncher.

The game is well packaged and presented and includes options for turning the sound on and off, control using keyboard or joystick and a hiscore table as well as a set of great rhymes describing each creature.

It costs £7.95 and will be available after Christmas and is highly recommended.

Cathie Burrell, Financial and Administrative Director was also a Director of Tansoft Ltd and is well known for her outspoken views on the way the software industry is organised and financed.

Geoff Phillips, is Software Director and is responsible for all software development and liason with the companies freelance programmers. He has written several books on the Dragon and Oric and is also the Author of several best-selling programs for the Oric.

The company has had substantial financial backing from private investors, a company in the U.K. and a company in France. The presence of a French Director, Marc Lafage, on the board will ensure that Orpheus has a high profile in the valuable and expanding French market.



ATMOS COMPATIBLE SOFTWARE BY DORMERE

Dormere Software's Orical Invaders and Orical Backgammon games were incorrectly shown in the last issue of Oric Owner's software list, as being available only on the Oric-1. This is in fact incorrect.

Dormere's software has always been compatible with the Oric-1 and Atmos, since the introduction of the Atmos.

All Dormere's software includes a 'Good Load' test before the program runs, thus ensuring that a corrupt load of the program is detected and the user informed. Well done Dormere.



BRUCE EVERISS JOINS TANSOFT



Bruce Everiss has been appointed Managing Director of Tansoft, the Cambridge-based software house specialising in the Oric computer.

In addition to the software business Tansoft will be distributing Oric hardware and has been contracted to carry out Oric's marketing function. It is also intended to open a small retail computer store at the Tansoft premises on the Cambridge Techno Park.

Bruce's first aims are to expand the Tansoft catalogue to cover other microcomputers and to establish the Oric Atmos in it's rightful market position. The Atmos delivers an incomparable price/performance ratio along with build quality and reliability that are vastly superior.

THE NEW ORIC IQ164

After much speculation, Oric has announced the pending release of the new Oric IQ164.

The price has yet to be announced and it is expected that A.S.N., France, will receive the first supplies.

Turn to pages 61 & 62 for full details of the new machine.

ORIC OPENS UP FINLAND



Oric Products International has further increased its presence in Europe by winning orders worth nearly £ $\frac{1}{2}$ million from Finland for its British-made Atmos 48K microcomputers, peripherals and software.

This follows the recent appointment of a new distributor in Finland by Oric

Products Export Limited – Comtron Finland.

Comtron has already supplied 1,000 Atmos micros to the Lohja chain of stores and is to provide a further 2,000 units, plus 250 Oric disc drives, 250 printers and software for the initial contract.

Reliability and comprehensive features were the key reasons for Lohja choosing the Atmos, which was evaluated in competition with the Acorn Electron and the Dragon.

The latest orders from Europe are the result of Oric's active campaign to tap the enormous potential offered by European markets for the Atmos. Oric has now been the home computer leader in France for over a year and last summer won \$2.75 million orders from Germany, Austria and Switzerland. This followed favourable reaction to the Oric designed 'German style' Atmos keyboard for German speaking users.

ATMOS POWER BUFFERS

The NIKE product range of power buffers safeguard computers against momentary and short duration mains failures. They are also effective against certain types of mains borne interferences.

NIKE AT is for use with the Oric

ATMOS

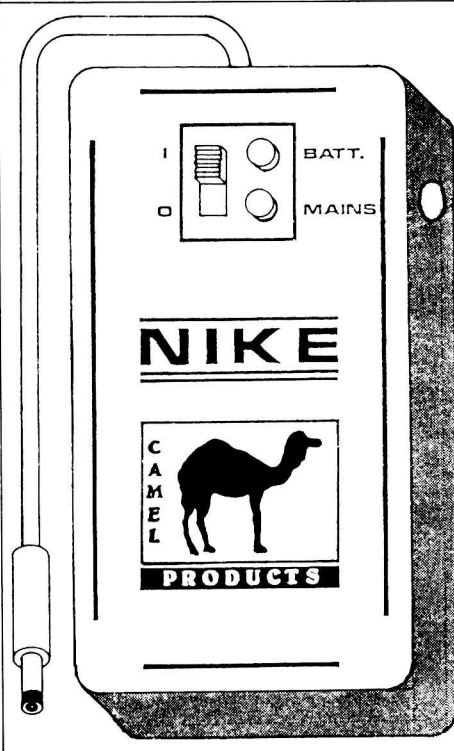
Rechargeable nickel cadmium cells are kept on floatcharge by the mains power pack. When the mains supply is lost, the cells take over the task of supplying the computer. Over 30 minutes of back-up is provided.

More rapid charging of run-down cells is achieved by keeping mains power on with the computer switched off. Manufacturers original power packs must be used.

NIKE SP supports Interface 1 but not microdrives.

The units include batteries and are ready to use. An ON-OFF switch for the computer is provided. Two coloured LEDs act as status indicators.

Helpful SAVEing techniques are given in the User notes.



ORIC/IJK JOYSTICK INTERFACE

Oric have reached an agreement with IJK Software whereby Oric will manufacture and sell the interface designed by Pennant and promoted by IJK. This interface plugs directly into the printer port and allows the connection of two Atari standard joysticks. Unlike the Pase interface the Oric/IJK design does not effect the sound in any way. It also

BACK ISSUES

If you've enjoyed this issue of Oric Owner, you may like to avail yourself of our back issues service. Some issues are sold out completely, but we do have limited numbers of issues 7 (April/May '84), 8 (June/July '84) and 9 (Oct/Nov '84). Back issues cost £1.20 each. Make cheques or postal orders payable to Tansoft Ltd.

April/May '84 – Practical Machine Code II, Pinball, educational software, Append/Renumber, Forth, hints, tips and much, much more.

June/July '84 – Dr Paul Johnson interviewed, Microdrives on test, sound explained, Star Cruiser, Slot Machine, Puss in Boots, Yahtzee, plus quickies, letters, hints and so on.

Oct/Nov '84 – Paul Kaufman interviewed, a look at the Oric factory, more of Practical Machine Code, a comprehensive list of software available for the Oric-1/Atmos, as well as software and book reviews and the regular features.

allows the games programmer to automatically check whether the joystick is connected.



All new IJK Software will support this interface and all new Tansoft Software has been converted to run with this new interface.

In addition Oric have launched their own joystick. This has a very high quality smooth action and is coloured to match the Atmos range.

The interface is priced at £11.50 and the joystick at £12.95.

Club Spot

As you have all now been told, we have incorporated the Official Oric Club into this magazine. We have had several letters from people in the U.K. and from abroad, asking us to mention their various clubs, which we are only pleased to do.

If any of you have started up a club and would like us to mention it, drop us a line.

Here are the contact addresses . . .

Belgium

Oric Club Verviers,
Mr Philippe Mawet,
72 Rue Wauory,
4502 Queue du bois,
Belgium.

Belgium

Mr Kassin,
Rue de L'Idylle 19,
1080 Bruxelles,
Belgium.

Belgium

Oric Computerclub Floppy,
p/a Mathieu Doumen,
Dennenlaan 7,
3460 Maasmechelen,
Belgium.

Belgium

T. Grosbois,
C.U.F.O.,
Rue de la Tourette 1,
5650 Biesme,
Belgium.

Belgium

Roger Hamblok,
Hoeverdijk 9,
3900 Lommel,
Belgium

Holland

Nedoric,
Dhr. G. Tax,
Gasthuisring 24,
5041 DS Tilburg,
Holland.

Germany

Holger Schlueter,
Lilienthalstrasse 16,
4460 Nordhorn,
W. Germany.

Germany

21 Engr Computerclub,
c/o Lt Col Beirne J.P.,
7FD Sqn RE,
21 Engr Regt,
BFPO 48.

Poland

R. Wacławek,
Młody Technik,
PO Box 380,
PL 00-950 Warsaw,
Poland.

UK

Cardiff

12 Tregarth Court,
Creigiau,
Cardiff.

Kent

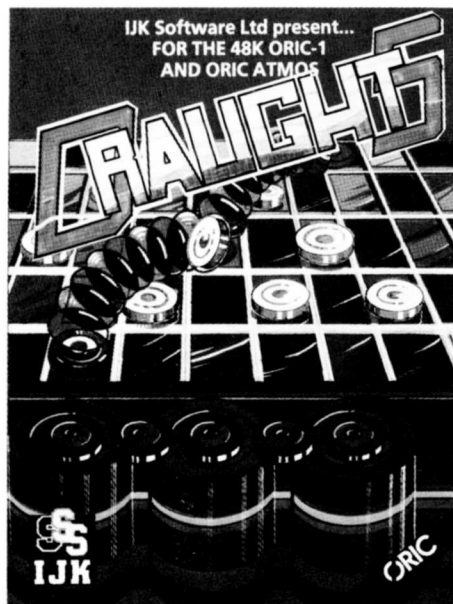
c/o Roger Pyatt,
23 Arundel Drive,
Orpington,
Kent.
Tel: 66 20281

Middlesex

Harrow Oric Owners Club,
c/o B. A. Crowe,
352 Lansbury Drive,
Hayes,
Middx.,
UB4 8SW.

Software Scan

Draughts
I.J.K. Software
Oric-1/Atmos
£7.50



This is a first for the Oric-1/Atmos from I.J.K. Software.

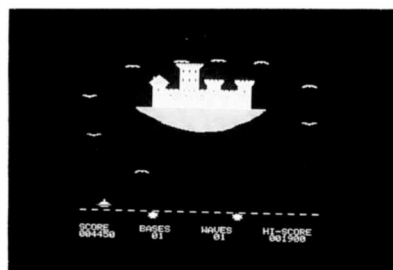
With the cassette you get a small booklet containing instructions. You have a choice of rules to follow which are: no restrictions, must take, must take all and must take most. This choice appeals to me as it means that the games you play are varied and you're more likely to keep playing.

A record and the time taken for each move is displayed at the side of the screen and the overall time at the bottom, a good idea for those of you wanting to increase your thinking speed.

It can either be played with two people or against the computer. If you choose the latter you might like to select the option you have of making the computer make the worst move.

Well worth investing in, if you like board games.

Originality	★★★
Graphics	★★
Addictiveness	★★★★
Value	★★★★



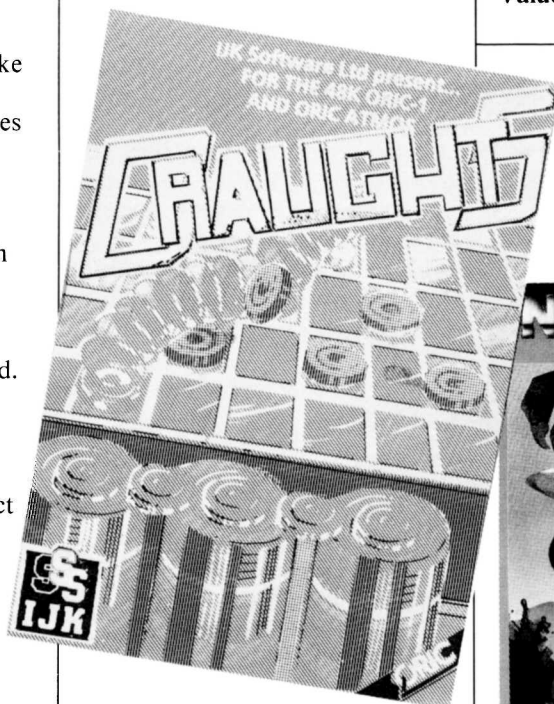
Styx
No Mans Land
Oric-1/Atmos
£8.50

This is one of a range of titles from the new French company No Mans Land.

The game itself is little more than a phoenix variation, though this said it is graphically very pretty. The swirling aliens above drop bullets and strange blobs onto your base at the bottom of the screen. If one of the blobs reaches the bottom, it lands and wanders about on the bottom line, attempting to stab you in the back. If you shoot the big green bird, all the blobs on the bottom disappear. After the first screen comes the flashing red aliens, and after that is a very pretty red castle to destroy.

Good, though the graphics and sound are, I got tired of the game very quickly.

Graphics	★★★★
Sound	★★
Addictiveness	★★
Value	★★★

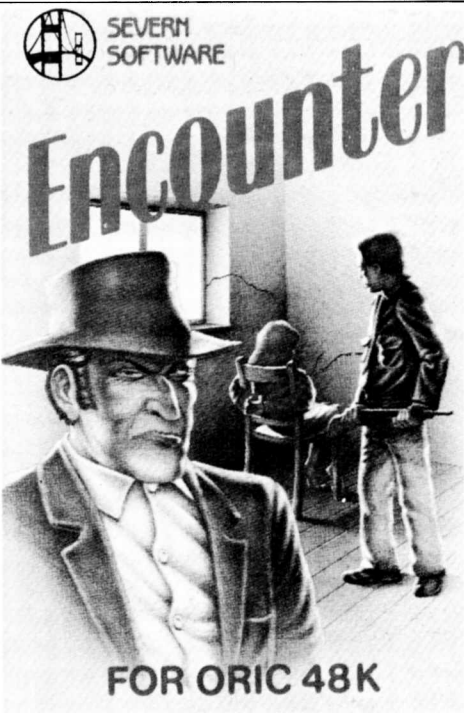
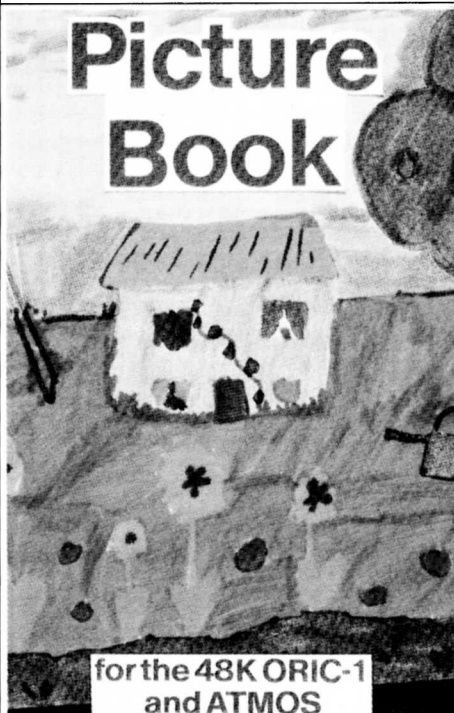
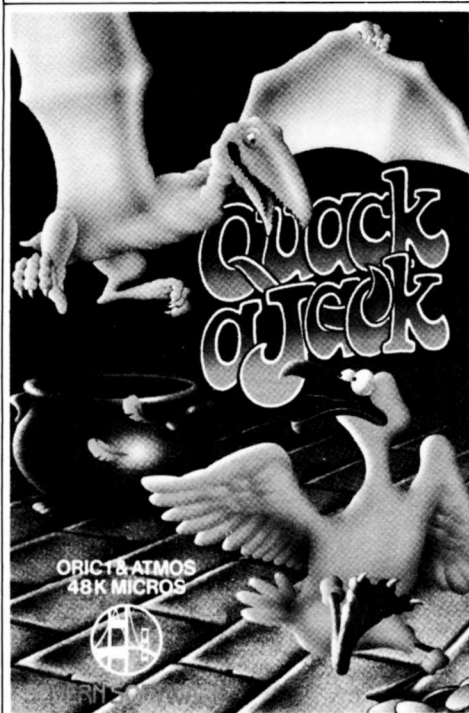


Software Scan

Quack-A-Jack
Severn Software
Oric-1/Atmos
£6.95

Picture Book
Softbacks
Oric-1/Atmos
£4.95

Encounter
Severn Software
Oric-1/Atmos
£7.50



Red Jack, the famous pirate duck, is trapped in the palace dungeons. You have to leap from flagstone to flagstone to get the Terraducktile before it hatches.

The graphics are good; Red Jack actually looks like a duck and it's easy to tell the difference between things to step on and things to avoid.

You must avoid the cooking-pots, which are scattered around the flagstones and go for the eggs before the coins (the dungeons inhabitants wages!) as it's important that you **do** stop the Terraducktile hatching, otherwise the whole screen stands still and all you can do is watch yourself die.

The only problem is unless you hold the keys down to move, it takes a long time for anything to happen. However, once it's warmed up, it doesn't move too badly.

It's not too difficult to gain a high score – points are very high when you step on the eggs and coins.

Although it was slow, I quite enjoyed this game. It's certainly a different idea and I found it quite addictive.

If you have young children and are on the point of pulling your hair out trying to find things to keep them occupied (and quiet!), then Picture Book is what you've been looking for.

Picture Book is an exciting new way of making pictures, it comes with an illustrated word list to select the drawings. There are 35 words and drawings set up in the program and a special option allows you to add extra words and drawings yourself (this is rather tedious). There is room for 150 words altogether.

Other options include: saving your drawings, turning the sound on and off, changing drawings and creating a file to store your new drawings.

A background of a few colours appears on the screen and the child is then free to fill in the picture, using its imagination. There are various backgrounds, so it doesn't get monotonous.

This is the best Oric educational software for young children yet and very good value.

Encounter is a classic-style text adventure.

Your task is to find a young girl, who has been kidnapped by thugs. Your clue is that a knowledge of Chemistry helps – I'm afraid that didn't help me much!

It is a good idea to make a map as you travel, noting what you find and where, as you may want to come back to it as the adventure progresses. It's also a good way of making sure you don't keep coming back to the same place.

The time limit is 500 moves. The number of moves you make are shown after each command typed in. Every time you pick something up you use one move, which is why I recommended that you make a map to pick things up only if you really need them.

There is a very limited vocabulary compared with many other text adventure games. It doesn't understand 'Look' which is usually a useful command.

At £7.50 I feel that it is overpriced and would advise you to buy one of the better adventures.

Originality	★★★
Graphics	★★★★
Sound	★★
Addictiveness	★★★★
Value	★★★★

Graphics	★★★★★
Sound	★
Interest	★★★★★
Value	★★★★★

Originality	★★
Addictiveness	★
Interest	★★
Value	★★

Software Scan

Gravitor Severn Software Oric-1/Atmos £7.50	Story Book Softbacks Oric-1/Atmos £4.95	DEBUG No Man's Land Oric-1/Atmos £11.95
		
<p>This game is a straight conversion from the arcade game of the same name. It is an adaptation on Asteroids, except that your ship is near the surface of a planet, so Gravity plays an important part.</p> <p>Your mission is to destroy a number of alien satellites, in three stages, to prevent an impending alien takeover in various parts of the universe. The three stages are Home Base stage, Satellite stage and Reactor stage.</p> <p>In the first part you steer your ship through a tunnel to reach one of four circles, representing the different parts of the planet surface, once there you negotiate past the UFO and Turrets that shoot at you to get close to four red squares on the land. These you pick up using your tractor beam. If you die you return to the first section, which can get a little tedious.</p> <p>There are seven levels of play in which sometimes some very delicate handling of your craft is required.</p> <p>Overall this is worth getting since it is one of those 'just one more go' type games.</p>	<p>Story Book can be used in a variety of learning situations: helping very young children to familiarize themselves with words, encouraging older children to compose stories and poems and even students of a foreign language will find this useful for practice and revision.</p> <p>There is a short example you can load, after the main program, to show you how the main program works. It is called Proverbs and involves guessing the right words and typing them in to complete the sentence. A sun smiles if you get them right and frowns if they are wrong – this part is geared towards young children.</p> <p>In the main program you are given four options, which are: recreate the story, alter the story, save the story on cassette or have a new story. You then have a choice of large or normal text. A blank screen appears and when you have chosen ink and paper colours, you can begin typing.</p> <p>Not quite as impressive as Picture Book but still a good educational program.</p>	<p>DEBUG is a monitor/program debugger, designed for writing and debugging Machine Code programs. The program, when loaded, takes up 6K starting from \$500. As this is where Basic programs are stored, it is not possible to have a Basic programs are stored, it is not possible to have a Basic program in memory whilst using DEBUG. This is a drawback if you are writing combined Basic and Machine Code programs.</p> <p>DEBUG has a range of commands for altering, comparing, displaying, filling and copying memory and performing other functions. Up to 10 breakpoints can be set and there are commands to single step a specified number of instructions or to step ignoring all jump routines.</p> <p>Although DEBUG provides a good range of useful and easy to use commands, it lacks a disassembler which is virtually essential for Machine Code programming.</p> <p>The program is well designed and if it had a disassembler it would be well worth buying it, however without this it is a bit limiting and you would be better off buying one of the monitors which does have a disassembler.</p>
Originality ★ Graphics ★★★★★ Sound ★★ Addictiveness ★★★★★ Value ★★★★★	Ease of use ★★★★★ Sound ★★ Interest ★★★★★ Value ★★★	Facilities ★★ Instructions ★ Ease of Use ★★★★★ Value ★★

Flight 401 Knight Software Oric-1/Atmos £7.50	The Diabolical Tower No Man's Land Oric-1/Atmos £8.50	Attack of the Cybermen I.J.K. Software Oric-1/Atmos £7.50
		
<p>Flight 401 from Knight Software is a flight simulator. Your task is to take off and land at one of a selection of airports. The display gives you all the vital information, altitude, air speed etc. However, the display is updated only once every two seconds. This means that even taking off is an exasperatingly slow process. Once in the air I was once more amazed at the appalling response time. The aircraft that this was modelled on must have the aerodynamic qualities of a wardrobe!</p> <p>Full flight instructions are included in the manual that comes with the tape and are also written into the simulator. However, this did not really help as it took so long to do anything.</p> <p>I finally managed to get the plane into a steep dive and crash, whereupon the computer told me that I had landed with the undercarriage up.</p> <p>There is an optional joystick control. An Atari type joystick with suitable interface is recommended. I'm afraid I didn't have the patience to try it again with a joystick.</p> <p>After playing it once, I had no incentive to go on.</p>	<p>The game starts with rather a lengthy tune, especially when you find that every time you die and want to start up again, you have to listen to it before you can go on. However, you can then choose whether you want sound effects or not. You also set a time for the text to be screened, so that you don't find you're waiting for ages for something to happen. You can choose whether or not to have pictures. This does not mean you are entirely without graphics, if you choose not to have them, as a floorplan appears without any question marks to show you where things are.</p> <p>Your mission is to find a fabulous treasure hidden somewhere in the tower. On your way around, you seem to come across a lot of shiny objects (which turn out to be gold coins) and boxes on tables (which usually have keys in them, although one I opened was a booby-trap and exploded in my face!)</p> <p>If you decide to stop or get discouraged, the message "Is there anyone near you who wants to play?" appears, which I found quite amusing. I found it impossible to stay alive and I recommend that you have a lot of patience if you buy this game.</p>	<p>This is IJK's latest release for the Oric-1/Atmos and is a variation on the arcade classic: Cybertron.</p> <p>In the 21st Century A.D. the existence of life on earth is threatened by the alien Cybermen.</p> <p>In the game you control a rather plump character called Percy. On the first screen you are surrounded by what I assume are the Cybermen, though they look more like sheep to me. You can move in four directions, firing all the time. The Cybermen close in and a flying donut attempts to catch you unawares. After this comes an original screen in which you try to reach home whilst avoiding the blobs thrown at you by a column of green men.</p> <p>It takes quite a while to master the game and even when you have, the deadly Bouncer seems to catch you unawares. It's not the sort of game that makes you want to give up – quite the opposite, I had to keep playing.</p> <p>The game supports three different joystick interfaces, including the new Oric one.</p> <p>All in all, an interesting variation on a well established idea.</p>
Originality ★ Graphics ★★ Sound ★ Addictiveness ★ Value ★	Originality ★★★ Graphics ★★★ Sound ★★★★★ Addictiveness ★★ Value ★★★	Originality ★★ Graphics ★★★★★ Sound ★★★★★ Addictiveness ★★★★★ Value ★★★★★

Software Scan

In the last issue we printed a long list of software available for the Oric-1 and Atmos; here are the addresses of most of the software houses.

Add-On Electronics Ltd., Units 2, 3, 4, Shire Hill Ind. Est., Saffron Walden, Essex.	Haresoft Ltd., PO Box 365, London NW1.	Mikro-Gen, 1 Devonshire Cottages, London Rd., Bracknell, Berks.	R & R Software Ltd., 5 Russell St., Gloucester.
Arcadia Software, 4 Sunningdale Ave, Swansea.	Honeyfold Software, Standfast House, Bath Place, High St. Barnet London, EN5 1ED.	Micrograf Ltd., PO Box 17, Bracknell, Berks.	Romik Software, 272 Argyll Ave., Slough, Berks.
Arctic Computing Ltd., Main Street, Brandes-Burton, Driffield, Yorks.	I.J.K., Unit 3C, Moorfields, Moor Park Ave., Bispham, Blackpool.	M.C.P., 13 High St., Clydach, Swansea.	Sector 7 Software, PO Box 8, Newton Abbot, Devon.
Basic Concepts, PO Box 46, Leatherhead, Surrey.	IMF, 143 Uxbridge Road, Ealing, London, W13 9AV.	Nectarine, 837 Yeovil Rd., Slough, Berks.	Severn Software, 15 High St., Lydney, Gloucestershire.
Bugbyte, 100 The Albany, Old Hall St., Liverpool.	Level 9 Computing, 229 Harpenden Rd., High Wycombe, Bucks.	Ocean Software, Ralli Building, Stanley St., Manchester.	Sigma Soft, 13 Westmorland Terr., Holmes Chapel, Cheshire.
Byteland, PO Box 57, Sawbridgeworth, Herts.	MC Lothlorien, 56a Park Lane, Poynton, Cheshire.	Peach Computers Ltd., 192 Greenock Rd., Largs, Ayrshire.	Softbacks, PO Box 257, Watford, Herts.
CDS Silver House, Silver St., Doncaster.	Melbourne House, Castle Yard House, Castle Yard, Richmond, Surrey.	Peaksoft, 48 Queen St., Balderton, Newark, Notts.	Softek International Ltd., 12-13 Henrietta St., Covent Garden, London WC2.
Darkstar (Superior), 2 Regent Crt., Amhurst Park, London N16 5LP.	Mellowsoft, 23 Dalford Court, Hollinswood, Telford, Shropshire.	P.S.S., 452 Stoney Stanton Rd., Coventry.	Tansoft Ltd., Units 1 & 2, Techno Park, Newmarket Rd., Cambridge.
Dormere Software, Belgreen House, Green St., Macclesfield, Cheshire.	Mercury Software, PO Box 21, Darwen, Lancs.	Quark Data, PO Box 61, Swindon, Wilts.	Taskset Ltd., 13 High St., Bridlington, Yorks.
Durell Software, Castle Lodge, Castle Green, Taunton.	Mr Micro Ltd., 69 Partington Lane, Swinton, Manchester.	Q.E.D. Systems, Liver House, 96 Bold St., Liverpool.	Touchstone, 61 Highfield Rd., Saxilby, Lincoln.
Epsilon Software, 10 Southern Road, West End, Southampton.	Microdeal Ltd., 41 Truro Rd., St Austell, Cornwall.	Quicksilva, Palmerston Park House, 13 Palmerston Rd., Southampton, Hants.	Virgin Games, 1-63 Portobello Rd., London W11.
Express Software, City Gate, 25 Moat Lane, Birmingham.			Wintersoft, 30 Uplands Park Rd., Enfield, Middx.



ADVANCED USER GUIDE

- * Fully commented ROM disassembly
- * Using routines in the ROM
- * Programming the sound chip
- * Intercepting interrupts
- * Interfacing details
- * Complete circuit diagram
- * Project to build a Real-Time-Clock



ADDER

Applicable to both the ORIC-1 and ATMOS, this book provides extensive information about the Oric computers. Written by Leycester Whewell of Cambridge University, the Advanced User Guide contains a Complete Commented ROM Disassembly (for the Atmos but with cross references to the Oric-1) plus lots of information on the sound chip, 6502, ULA, BASIC variable storage, intercepting interrupts, the disc system and a complete circuit diagram. Full interfacing details including a project to build a Real-Time-Clock are also included.

ISBN 0 947929 01 0, 288 pp, £8.95 nett

Available from larger retail outlets or by mail order from:-
ADDER PUBLISHING Limited, P.O. BOX 148, CAMBRIDGE, CB1 2EQ

Please send cheque or PO for £10.45 (inc. p&p).

Softbacks

Educational software . . .

Story Book - A challenging game that develops reading skills. Suitable for any level from early reader onwards. You choose the text, the game is to reconstruct it by guessing the words. £4.95

Picture Book - A colourful spelling game for young children. Build up pictures by typing words and seeing the objects appear "by magic". £4.95

Personal finance . . .

Account Book - The money management program. Budgeting and record keeping by automatic double entry. Produces full reports on screen or printer. Easy to use and menu driven throughout. £12.50

For further details phone (0923) 53482 or write to:

Softbacks, FREEPOST, Watford, WD1 8FP.

No stamp required if posted in the U.K.

Please state model of computer when ordering.

DEDUCT 50p from total for more than one program.

For the ORIC-1 and ATMOS

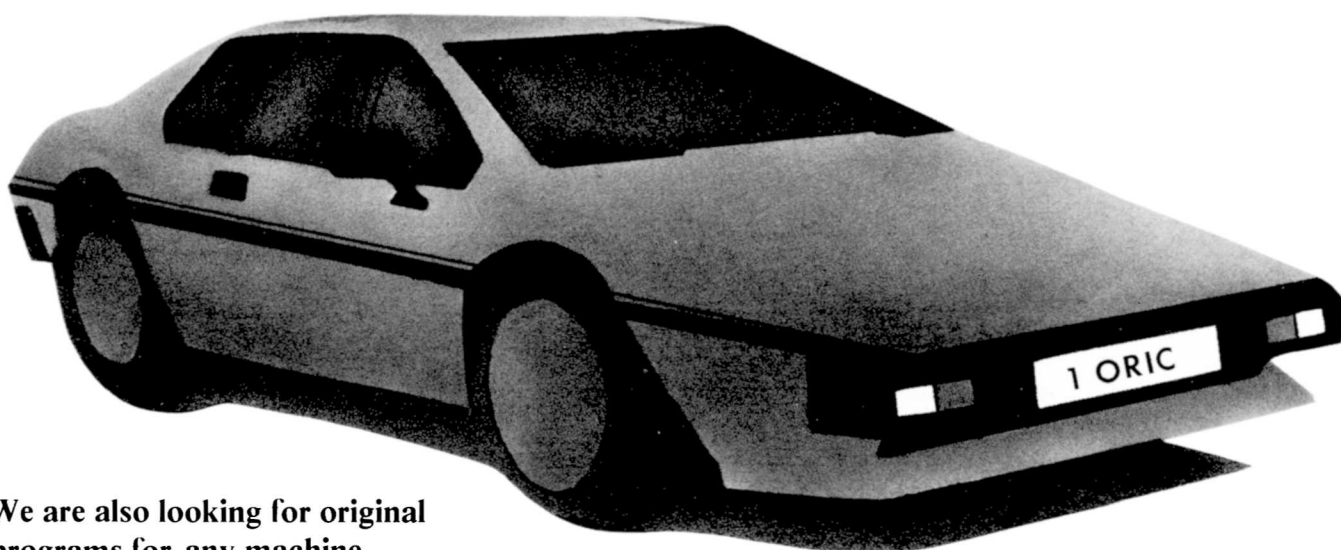
MAKE MONEY

Competent at assembly language programming?

We need software converting from other micros to the Atmos and vice versa.

Our fees won't make you a Lotus driving whizz kid, but you will make some money from your hobby.

Contact: Adrian Rushmore,
Tansoft Ltd.,
Unit 1,
Techno Park,
Newmarket Road,
Cambridge, CB5 8PB
Tel: (02205) 2261/2/3/4



We are also looking for original programs for any machine.

Christmas Carols

A.Thom

This program is designed to get you all in the Christmas spirit. Mr Thom has written five different carols in three part harmony which will run on any Oric.

They are quite loud when run, so if you've got any relatives coming round over Christmas that you want to keep quiet, have it ready!

The five carols are: "O Come All Ye Faithfull", "O Little Town Of Bethlehem", "The Coventry Carol", "While Shepherds Watched" and "Once In Royal David's City", so you shouldn't have any trouble with the words if you decide to sing along.



FOOTBALL POOLS

E.Hollister

The program detailed below will provide forecasts of football league matches (1984-85 season) from the four English Divisions.

Forecasts are made on the basis of team form information which is keyed in by the user. Emphasis has been placed on making the program quick and easy to use, the design contains the following features:

1. A menu driven system with four choices.
 - i) Updating the form data
 - ii) Obtaining the forecasts
 - iii) Saving updated form
 - iv) Terminate program and reloading notes



One may argue that if everybody is using the same system then they are all getting the same forecasts, however, the program is easily modified to suit individual tastes. There are two ways of doing this:-

1. The values allotted for results may be adjusted, see lines 1460 to 1510.
An away win (AW) might be worth more than '5' or a home draw (HD) more than '1', etc.

2. The window for setting a draw may be widened or shifted, see lines 1980 to 2010.

W holds the form data of the home team (Page 4 addresses hold '85' to start with, 75 is deducted to give a working base of 10. This prevents going over the maximum value of 255 for an address in the later stages of the season, but keeps away from zero, where dividing would be invalid).

X and Y set a value for 'Home advantage' and the lower edge of the 'Draw' window.

Z sets the upper edge of the 'Draw' window, any value above Z represents an 'Away'.

In line 1990 dividing W by a number larger or smaller than 10 will shift the 'Draw' window down or up respectively.

In line 2010 the 'Draw' window may be narrowed or widened by decreasing or increasing the number of X's (X's can be multiplied instead of adding).

A typical sequence of actions for using the program is shown below.

1. Load the Form Data tape.
2. Type NEW and Return.
3. Load the Main program tape.
4. Run the program, the menu will appear.
5. Select and use the Updating Form routine.
6. Select and use the Obtaining Forecasts routine.
7. Select and use the Save Updated Data routine.
8. Select End routine.

NOTE: Form Data is stored in page 4 addresses #0401 to #045C (92 addresses). This is done automatically under program control in choice 3 of the menu.

The program can be modified to cover Scottish or Australian leagues by adjusting the team names in the data statements and the number and length of the divisions.

For use in future seasons data statements must be adjusted to suit promotion and relegation changes.

The program was written on an ORIC-1 48K machine and uses 15.9K of memory.

ETCH-A-SKETCH

P. COOPER

Etch-A-Sketch is a computer based equivalent of the well known drawing pad. One or two enhancements have been added to make it in to a more versatile drawing aid, i.e. memory screen, 'rubber banding', m/c fill routine, etc. . .

A ring of keys are used to control the position of the graphics cursor on the screen. Details are given from within the program for direction of movement, but the keys used are O P | K ^ , . and / enabling eight directions of cursor movement. The X and Y positions of the cursor are given at the bottom of the screen.

The ten numeric keys are used to select different modes as follows. . .

KEY 1 UNPLOT

Cursor changes to background colour. The pixel that the cursor is positioned on is set to background colour.

KEY 2 PLOT

Cursor changes to foreground colour. The pixel that the cursor is positioned on is set to foreground colour.

KEY 3 MOVE

Cursor flashes. Pixels are not affected.

KEY 4 DRAW

('Rubber Banding'). Line of pixels flash, between current cursor position and last two cursor positions, in preparation for 'Fixing' in foreground colour. See 'Mark' and 'Fix'.

KEY 5 ERASE

('Rubber Banding'). Line of pixels flash, between current cursor position and last two cursor positions, in preparation for 'fixing' in background colour. See 'Mark' and 'Fix'.

KEY 6 COLOUR

Plots attribute at current cursor position (cursor stops flashing in 'move' mode when positioned on an attribute). Attribute selected by pressing key of relevant escape code. See Appendix 2, Atmos manual page 232.

e.g. To plot green ink at current cursor position, press '6'

to select colour mode, then 'B' to select green ink. To plot inverse attributes press '6' twice before selecting attribute.

KEY 7 FILL

m/c fill routine to fill enclosed area with foreground colour, starting at the current cursor position. Ensure area is completely surrounded otherwise the fill routine will overflow onto the rest of the screen. The routine fills all but the most complicated shapes and takes about 15 seconds.

KEY 8 CIRCLE

Draws a circle around current cursor position in foreground colour. Expects 2 digit number to determine radius of the circle (01-99) in pixels.

KEY 9 SAVE

Saves HIRES screen to tape at fast speed.

KEY 0 LOAD

Loads HIRES screen from tape at fast speed.

Two more keys are used in the draw and erase modes. These are . . .

KEY 'ESC' MARK Marks current cursor position.

'SPACE BAR' FIX

Fixes line between current cursor position and last two marked position.

In addition, three keys are used to provide a memory function. These are . . .

KEY '-' COPY

Transfers hires screen to memory.

Key '=' SWOP

Exchanges HIRES screen with memory

KEY ' ' CLEAR

Clears memory

also . . .

'DEL'

Clears the HIRES screen.

PITFALL

P. Turner

This program is a variation on the Donkey Kong theme. It involves climbing ladders and jumping over obstacles.

You are trying to escape from a cave structure by climbing up rope ladders left by previous expeditions. Rock falls cause boulders to roll down the passages unless you jump over them.

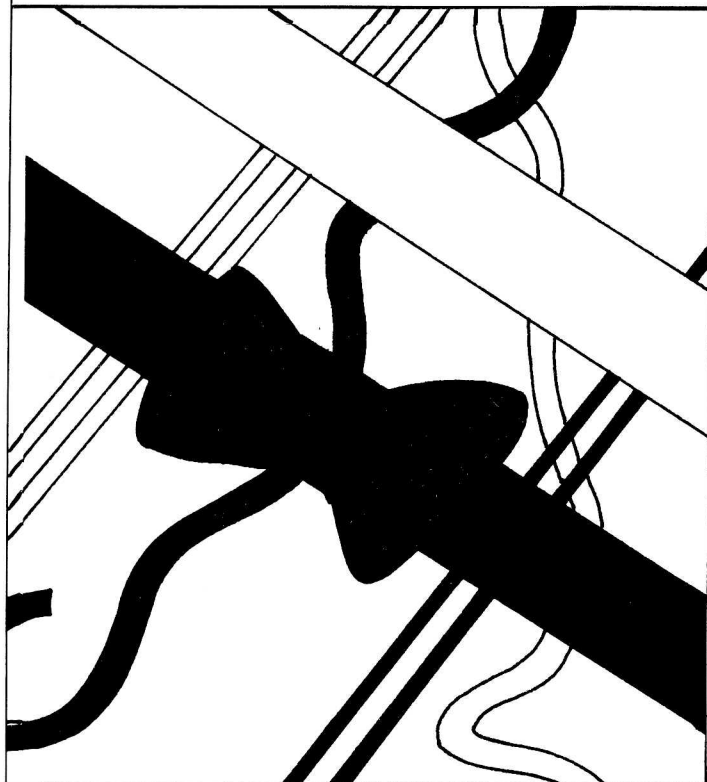
You can move at will up and down the various levels represented by different colours from the pit to the outside world.

It will only run on the Oric-1 48K and the controls are:

← = left ↓ = down
→ = right ↑ = up
Space Bar = Jump



REVENGE OF THE BEANS



For years now the beans of Hinez have suffered untold cruelties at the hands of the human people, but now the time for their revenge has come. They have built giant nuclear driven bean tins and have flown to earth with the baking of humans in their minds. The beans jump from the cans in a Kamikaze style and you have to splat them before you are splatted.

From time to time the super can will fly across the screen dropping a continuous flow of beans.

A bean hit gets you 2 points and a super can hit is worth 20. Use:

left cursor move left
down cursor move right
FUNCT fire phasor

This game, written by D. Peat, will only run on the Atmos, as it uses specialized ROM routines. It comes as three separate programs. Type in the first program and run it saving the data on cassette, do the same for the second and third programs then to run it type, CLOAD"" :CLOAD"" : CLOAD""

midnight feast

D. REID

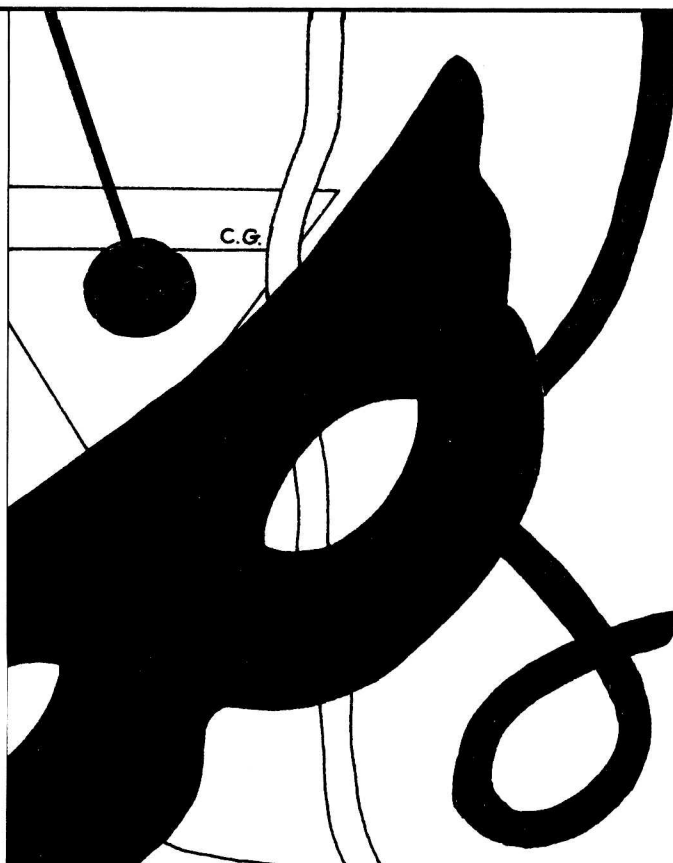
It's past midnight and you are feeling hungry. You must sneak around the house eating as much food as you can without waking up Mum and Dad. If you make too much noise they will wake up and you will be caught. Be careful not to stand on any creaky floorboards, and avoid the cat because it will "mieow" if you touch it!

You have £5 pocket money to begin with, and you lose £1 each time you are caught, as punishment! You start off on the ground floor and must make your way back upstairs to bed before the alarm goes off in the morning. There are various types of food scattered through the house and you will gain more points for eating bananas and pears than for eating biscuits etc.

When you get back to bed, a bonus is added to your score depending on the time left. Then you can try again the next night, which is more difficult since there are more creaky floorboards and the cats move faster. However there is more food available to be eaten!

The game ends when you have lost all your pocket money, and if you do well enough you can enter your name in the hi-score table.

The game will run on the Oric-1 and Atmos.



CRIBBAGE

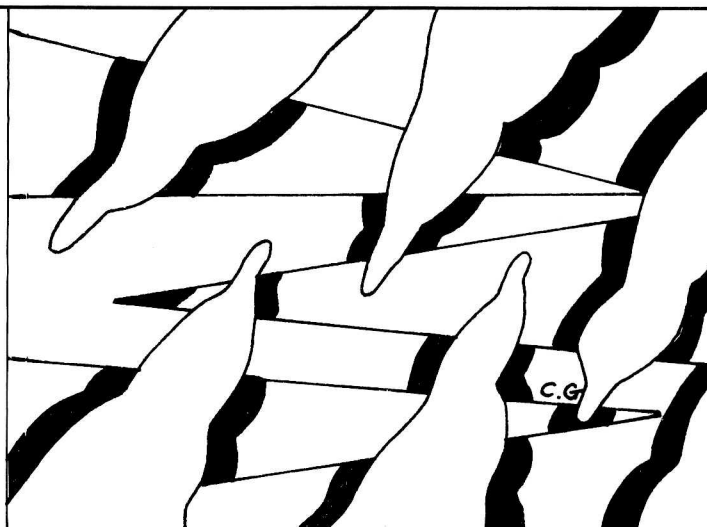
L. Rooke

This version of Cribb is very nicely presented and slightly different to the one I'm used to playing. The instructions and details of point scoring are very detailed though, so you should soon come to terms with the game.

It opens with a burst of music and you can choose if you want the computer to ping for pegging or not. I'd advise against it if you plan to play for a while – it does tend to get on your nerves.

Cribbage will run on both Oric-1 and Atmos machines. There are five levels of play in which you play against the computer. Surprisingly enough, the computer doesn't always win, as might be expected.

Don't be put off by the length of the listing – it's worth it!



Atmos Basic Utilities

D. Sinfield

Number 1: Auto line number

Most computers have Auto Line Numbering available in toolkits if not provided in the firmware. The poor old Atmos seems to have been left out in the cold yet again. At first look the program should be easy enough to write. But the V1.1 ROM conspires against any attempt to use its main loop that looks after getting keys, putting the ASCII code of the keypress in the input buffer, tokenising the input and slotting program lines into the correct place in the program ends in an absolute jump back to a position in the ROM thereby 'loosing' any controlling program. There are several ways of overcoming this difficulty. I have gone for the most straightforward – the relevant section of the ROM program is copied down into RAM where it can be altered to fit in with what we want it to do. Lines 100 to 155 of the BASIC listing do just this. The remainder of the program reads the original part from the data statements at 10000 onwards. A complete

Assembler listing is also given for those interested in how it all works. I think the annotations make it all quite clear.

How to use the utility

The BASIC should be entered, CSAVED (just in case), thoroughly checked and RUN. I have included a checksum to detect errors in the DATA statements.

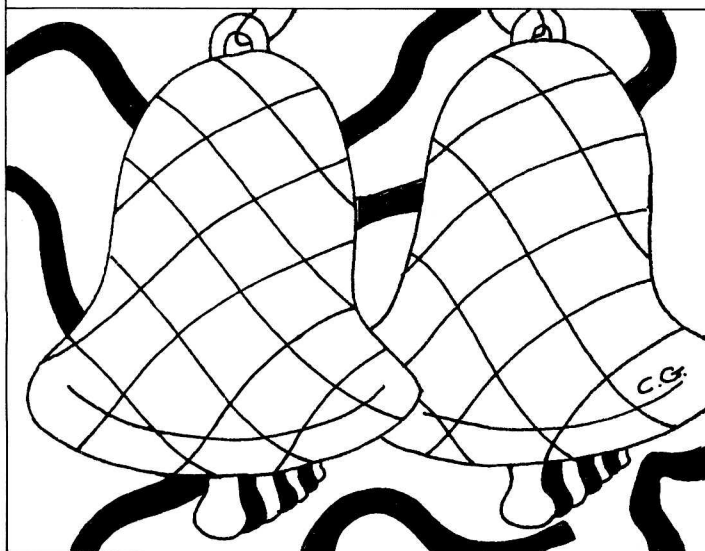
Assuming all is well the BASIC can be NEWed and the code from #954C to #96FF can be CSAVED for future use. The code is vectored to the ! command and the format of the instruction is:

! LLLLL,III

where the L's are up to five digits of starting line number and the I's are up to three digits of increment. There is substantial error checking but entering a line number out of range (more than 63999) is possible and it won't be rejected until return is pressed at the end of the line when a Syntax Error will be generated. Escaping from Auto is by using control C.

Beware of 'skipping' lines using the return key. It will look to the machine as though an empty line has been entered and if there was a line with that number it will be deleted. It would have been fairly easy to screen for empty lines but I use the program to block delete parts of programs.

The BASIC protects the program (HIMEM =945B) and sets up the ! vector (DOKE=2F5,=945C). If the code is loaded direct from tape, which is much quicker than via the BASIC, it is vital that these two instructions are entered. The code is located as high as my assembler would allow and as it allows plenty of room for both BASIC and variables I haven't relocated it right to the top. If you want to move it the assembler listing will help you pick out the absolute jumps that will need changing.



Atmos Basic Utilities

D.Sinfield

Number 2: Creation of Data Statements

Although it is easier on the Atmos to load blocks of memory on the Atmos than it is on the Oric-1 it is often more convenient to have machine code routines, character sets and even text screens stored as data statements within a BASIC program. The routine given here converts a given block of memory into DATA statements for use in BASIC programs. To the computer it appears that the data statements have been entered in the normal way via the keyboard and the result is a collection of DATA statements holding hexadecimal numbers. The controlling program NEWs itself so all that is left are the DATA statements it creates.

Calling the routine without this initialisation creates a line number 0.

The first thing I do with machine code routines I see in magazines is enter and disassemble them to see if I could have done it any better. To save anyone like me the effort I have included a disassembled listing.

Number 3: Interrupt counter

The Atmos generates interrupts every 100th second or so and this is when it usually flashes the cursor and looks for key presses. The return from the interrupt is vectored via page two and by diverting the program at this point a routine can be executed at regular intervals. This program is almost a clock but it seems that the interrupts aren't exactly 100th second apart because the clock runs a bit slow.

The machine code consists of two programs: an initialisation routine to set up and zero the clock and the program called at each interrupt that updates the clock.

Getting it started

I have included an assembly listing as well as the program reduced to data statements with a BASIC loader. You can use whichever you like to get the program into memory. There are no absolute jumps or calls to subroutines so where the program is positioned in memory is up to you. Whichever means of getting the program into memory you use, the code should then be saved.

When the code is in the clock has to be initialised by calling the set up routine. The exact address for the call will depend where in memory you have located the code but the way I have set it up the following instructions apply:

Run program

Initialise or set to zero: CALL#942C

Start clock: DOKE #24B,#942C:POKE#24A,#4C

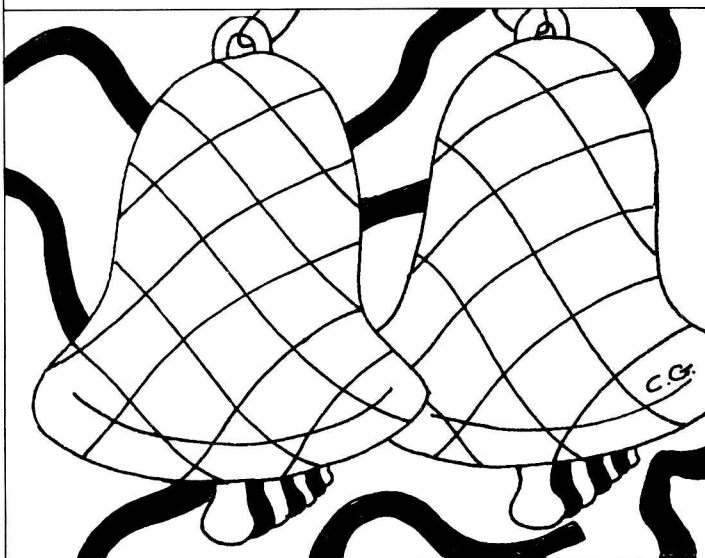
Stop clock: POKE#24A,#40

For subsequent starts only the POKE is required.

What to use it for

Because the counter runs whatever BASIC is doing it can be useful when trying to optimise your BASIC programs for speed. It can also be used in BASIC programs with time limits. The values can be PEEKed off the top of the screen, handled using the string handling etc. Although the clock is inaccurate it is useable over short periods (up to about 15 mins) as a 100th second stopwatch. It is entirely feasible to adjust the timing to make it absolutely accurate but I'll let you work that one out.

If you want to use the clock but don't want the numbers displayed POKE=BB81,0 will blank the display of the numbers.



The BASIC

The BASIC program first of all sets up the code. Lines 110 to 170 are from the Auto line number program. It may have seemed more sensible to combine the routines and then this copying down would only have to have been tackled once for two utilities. In fact the programs will rarely if ever be used together so I have made them totally independent of one another. Lines 180 to 230 READ and POKE the data for the routines that generate the line numbers, put the required bits in the input buffer etc. The remainder of the BASIC allow you to tailor the machine code to get different blocks of code and put them into the line numbers to suit your programs. The code at =4000 onwards can be used independantly of the BASIC but the start, finish line number and increment parameters have to be set up using the following pokes before calling the code:

The starting line number as ASCII codes is held in =404B to =404F

The increment goes into =4051 to =4053 as the digits of the increment

The Start of the block of code is doked into =4062

The end address + 1 of the code is doked into =4064

INDEX

<p>MIDNIGHT FEAST Fumble your way through the dark to the food without tripping over the cat. – Page 20</p>	
	<p>CRIBBAGE If you enjoy a good game of Cribb, this version is certainly worth typing in. – Page 32</p>
<p>REVENGE OF THE BEANS The beans from Hinez have set out to destroy the Universe. Can you stop them? – Page 24</p>	
	<p>PITFALL This game is written on the Donkey Kong theme. Climb the ropes and survive the falling rocks. – Page 43</p>
<p>CHRISTMAS CAROLS “Christmas is coming, the geese are getting fat”, why not sing along to these five well known carols? – Page 27</p>	
	<p>FOOTBALL POOLS Your Oric will help you to make those vital decisions that could win you a fortune on the Football Pools. – Page 45</p>
<p>ATMOS BASIC UTILITIES Three useful utility programs for your Atmos. – Page 30</p>	
	<p>ETCH-A-SKETCH Test your artistic skills with this program. – Page 48</p>

HOW TO SUBMIT PROGRAMS

We are always delighted to see your programs for possible publication in Oric Owner. If the program is less than 50 lines long, we will probably consider it a Quickie.

As we are kept busy putting the magazine together, we don't have time to type in long programs. If you are sending in a long program then, please send it on cassette. To aid correct loading, we would advise CSAVEing two listings, one at slow data transfer speed.

We have our own printers, to give a printout suitable for use in the magazine. It is therefore not essential that you send

a listing for BASIC programs. We must have listings for Machine Code programs though.

If you can include any details on special routines in your programs, or a brief summary of how they run, it would be helpful.

Finally, unless you include a stamped, self addressed envelope, we cannot return any submitted programs, due to the number we receive each week. Make sure you have a copy of your program at home, or enclose a s.a.e. if you want the cassette returned.

MIDNIGHT FEAST

```

0 GRAB:HIMEM #B3FF:POKE #26A,10

1 REM PROGRAM : Midnight Feast
2 REM AUTHOR : David Reid
3 REM MACHINE : Oric-1 or Atmos

5 CLS:PAPER 0:INK 6
6 PRINTCHR$(20)"Storing data .."
10 GOSUB 2000 'Initialize values
20 GOSUB 3000 'User graphics
30 GOSUB 3200 'Store machine code
40 GOSUB 2700 'Hi-score table
50 GOSUB 2000 'Reset values
60 GOSUB 2100 'Print title screen
70 GOSUB 3500 'Play tune
80 GOSUB 2300 'Draw house
90 GOSUB 2500 'Draw food
100 DOKE CLOCK,10000:C=0 ' start clock
k
105 PLOT0X,OY,"J":PLOT0X,OY-1,"I"
107 :
108 REM the main loop
109 :
110 REPEAT
120 IF NOT JU THEN 130
125 K$=KEY$:NX=OX-1-R*2:NY=OY+1:GOTO
180
127 :
128 REM what key is being pressed ?
129 :
130 K=ASC(KEY$+" "):POKE KE,4
135 IFK<>32THENSOUND1,OY*3,6:PLAY1,0,
1,1:PLAY0,0,0,0
140 NX=OX+(K=8)-(K=10)
150 IF K<>10THEN160
152 PLOT OX,OY,74:PLOT OX,OY-1,73
154 IF R THEN 200 ' facing right
156 R=TRUE:GOTO440
160 IFK<>8THEN170
162 PLOT OX,OY,68:PLOT OX,OY-1,67
164 IF NOT R THEN 200
166 R=FALSE:GOTO440
170 IFK<>11THEN 440 ' update clock
175 JU=TRUE:NX=OX-1-R*2:NY=OY-1:S=S+5
:PLOT9,2,STR$(S)
180 PLOT0X,OY,66-R*6:PLOT0X,OY-1,65-R
*6
197 :
198 REM check new position
199 :
200 OS=NS:OH=NH
210 NS=SCRN(NX,NY):NH=SCRN(NX,NY-1):F
L=SCRN(NX,NY+1)
215 JU=((FL=32)OR(FL=77)OR(FL=78))
220 OK=((NS<>77)AND(NS<>78)AND(FL<>83
)AND(NS<95)AND(FL<95))

```

```

227 :
228 REM move man
229 :
230 IFNS=79THENNXX=OX:NY=OY:NS=OS:NH=O
H
240 TS=SCRN(OX,OY):TH=SCRN(OX,OY-1)
250 PLOT0X,OY,OS:PLOT0X,OY-1,OH
260 PLOTNX,NY,TS+2:PLOTNX,NY-1,TH+2
270 OX=NX:OY=NY
280 IFNH<89THEN330
287 :
288 REM eat some food !
289 :
290 S=S+(NH-88)*20:NS=32:NH=32
300 PLOT 9,2,STR$(S):PLOT 30,26,CHR$(
6)+"yum "
310 FOR I=70 TO 10 STEP -5:SOUND 1,I,
6:SOUND 2,I/2,7:PLAY3,0,1,1000
320 NEXT I:PLAY 0,0,0,0:PLOT 30,26,0
330 IF OK THEN 440
340 IF NS>94 THEN 1900 ' reached bed
347 :
348 REM you've hit something
349 :
350 IF FL=83 THEN 400 ' creaky floor
355 IF FL>94 THEN 1900 ' new frame
360 PLOT 30,26,CHR$(6)+"mieow! "
370 FOR I=100 TO 250 STEP 15:SOUND 1,
I,8:PLAY 1,0,1,10:NEXTI
380 FOR I=250 TO 60 STEP-8 :SOUND 1,
I,8:PLAY 1,0,3,10:NEXTI
390 PLAY 0,0,0,0: GOTO 3700 ' caught
400 PLOT 30,26,CHR$(6)+"creak! "
410 FOR I=500 TO 200 STEP-10:SOUND 1,
I,10:PLAY 1,0,1,1000
420 PLAY 0,0,0,0:NEXT I
430 GOTO 3700
437 :
438 REM ok, update clock and counter
439 :
440 TX=INT(DEEK(CLOCK)/100)
450 IFTX>=INT(T)THEN550
460 T=TX:PLOT30,2,STR$(T)+" "
470 IFT>10THEN550
480 INK6
490 IFT>0THEN550
500 PLOT30,26,CHR$(6)+"rrring!":INK4
510 FORI=1 TO 30:PLAY 0,0,0,0
520 FORJ=1TO9:NEXT J:PING: NEXT I
530 GOTO 3700 ' caught you
547 :
548 REM updated ok
549 :
550 C=C+1:IFC<5-NTHEN 600
560 C=0:CALL #400:REM move the cats
570 IF((SCRN(NX,NY)=77)OR(SCRN(NX,NY)

```

MIDNIGHT FEAST

```

=78)) THEN NS=SCRN(NX,NY):GOTO360
600 UNTIL FALSE
1897 :
1898 REM new frame
1899 :
1900 T$="123456789ABCD&1&D&1&&":LT=LE
N(T$):TE=1: OC=4
1910 GOSUB 3540:INK 4:BO=T*10*N
1920 S=S+BO:PLOT9,2,STR$(S)
1930 N=N+1:PLOT 20,2,STR$(N)
1940 PLOT 15,14,CHR$(3)+CHR$(8)+"BONU
S"+STR$(BO)+CHR$(9)+CHR$(4)
1950 IF INT(N/7)=N/7 THEN M=M+1:PLOT
19,26,STR$(M)+".00":PING
1960 PLOT 0X,0Y,NS:PLOT 0X,0Y-1,NH
1970 GOSUB 2010 ' reset values
1980 GOSUB 3500 ' play tune
1990 GOSUB 2600 ' floorboards
1992 PULL
1995 GOTO 90 ' play this frame
1997 :
1998 REM initialize values
1999 :
2000 S=0:N=1:M=5
2010 T=99
2020 0X=8:0Y=23:NX=0X:NY=0Y
2030 OS=32:NS=OS:NH=OS:TS=OS:TH=OS
2040 JU=FALSE:R=TRUE:OK=TRUE
2050 FL=79:K=ASC(KEY$+" ")
2060 REM constants
2070 CLOCK=#276 ' internal clock
2080 KEYTM=#20E ' key repeat timer
2090 RETURN
2097 :
2098 REM print title screen
2099 :
2100 CLS:PAPER 0:INK 6
2110 A$=CHR$(10)+"Midnight Feast":PLO
T 11,1,A$:PLOT 11,2,A$
2120 PLOT 13,4,CHR$(3)+"By D.Reid"
2130 PLOT 10,6,CHR$(4)+CHR$(96)+" cop
yright 1984"
2140 FOR I=9 TO 13 STEP 2:PLOT 2,I,9:
PLOT 4,I,8:PLOT 1,I,3
2150 PLOT 22,I,9:PLOT 24,I,8:PLOT 5,I
,I-7:PLOT 21,I,3
2160 PLOT 25,I,I-7: NEXT I
2170 PLOT 6,9, "biscuit 20":PLOT 3
,9,89
2180 PLOT 6,11, "cheese 40":PLOT 3
,11,90
2190 PLOT 6,13, "cherries 60":PLOT 3
,13,91
2200 PLOT 26,9, "cocoa 80":PLOT 2
3,9,92
2210 PLOT 26,11,"banana 100":PLOT 2

```

```

3,11,93
2220 PLOT 26,13,"pear 120":PLOT 2
3,13,94
2230 PLOT 14,16,CHR$(1)+"CONTROLS"
2240 PLOT 10,18,CHR$(3)+"left arrow
LEFT"
2250 PLOT 10,20,CHR$(5)+"down arrow
RIGHT"
2260 PLOT 12,22,CHR$(6)+"up arrow JU
MP"
2270 PLOT 4,25,CHR$(12)+"Press <space
> to start game .."
2280 K$=KEY$:REPEAT:RN=RND(1):UNTIL K
EY$=""
2285 PLOT 4,25,0
2290 RETURN
2297 :
2298 REM draw house
2299 :
2300 CLS:PAPER 0:INK 0
2310 PLOT 4,26,CHR$(6)+"Just a sec ..
"
2320 FL$="PPPP": FOR I=0 TO 24
2330 PLOT 2,I,9:FL$=FL$+"P":NEXT I
2340 FOR I=5 TO 37:PLOT I,24,79:PLOT
I,4,80:NEXT I
2350 GOSUB 2600 ' draw floorboards
2355 REPEAT:CALL#400:UNTIL(PEEK(#4AB)
=#17):CALL#400
2360 FOR I=9 TO 33:PLOT I,0,86:NEXT I
2370 FOR I=0 TO 4: PLOT 8-I,I,84:PLOT
34+I,I,85
2380 PLOT 36-I,9+I,81:PLOT 6+I,14+I,8
2:PLOT 36-I,19+I,81
2390 PLOT 5,4+I,79:PLOT 37,4+I,79:PLO
T 5,9+I,79
2400 PLOT 37,9+I,79:PLOT 5,14+I,79:PL
OT 37,14+I,79
2410 PLOT 5,19+I,79:PLOT 37,19+I,79:
NEXT I
2420 PLOT 34,1,9:PLOT 34,2,9:PLOT 8,1
,8:PLOT 8,2,8:PLOT 35,2,4
2430 PLOT 6,7,95:PLOT 6,8,96:PLOT 7,8
,97:PLOT 8,8,97:PLOT 9,8,98 ' bed
2440 FOR I=4 TO 22 STEP 5:PLOT 3,I+1,
6
2450 PLOT 3,I+2,3:PLOT 3,I+3,5:PLOT 3
,I+4,1: NEXT I
2460 INK 4:PLOT 9,1,"Score Night
Time"
2470 PLOT5,26,"Pocket Money _":PLOT 9
,2,STR$(S):PLOT 20,2,STR$(N)
2480 PLOT 30,2,STR$(T):PLOT19,26,STR$
(M)+".00"
2490 RETURN
2497 :

```

MIDNIGHT FEAST

```

2498 REM draw food
2499 :
2500 FOR I=12 TO 30:PLOT I,6,32:PLOT
I,11,32
2510 PLOT I,16,32:PLOT I,21,32:NEXT I
2520 FOR I=1 TO N*4+15
2530 R1=INT(RND(1)*18+12):R2=INT(RND(
1)*4)*5+6
2540 IF((SCRN(R1-1,R2)>32)OR(SCRN(R1+
1,R2)>32)) THEN 2530
2550 PLOT R1,R2,INT(RND(1)*6)+89
2560 SOUND1,R1+R2,9:PLAY1,0,1,10
2570 PLAY 0,0,0,0: NEXT I
2580 GARBAGE=FRE("")
2590 RETURN
2597 :
2598 REM draw floorboards
2599 :
2600 PLOT 6,19,FL$:PLOT 8,14,FL$:PLOT
6,9,FL$
2610 PLOT 7,24,83:PLOT 7,19,83:PLOT 3
5,24,83:PLOT 35,14,83
2620 FOR I=1 TO N*2+6
2630 R1=INT(RND(1)*18)+12:R2=INT(RND(
1)*3)*5+9
2640 IF((SCRN(R1-1,R2)>80)OR(SCRN(R1+
1,R2)>80)) THEN 2630
2650 PLOT R1,R2,83
2660 NEXT I
2670 RETURN
2697 :
2698 REM set up hi-scores
2699 :
2700 DIM HI(8),HI$(8)
2710 FOR I=1 TO 8
2720 :HI(I)=(11-I)*200:HI$(I)="Oric"
2730 NEXT I
2740 RETURN
2747 :
2748 REM swap hi-scores
2749 :
2750 Z=HI(I):HI(I)=HI(I+1):HI(I+1)=Z
2760 Z$=HI$(I):HI$(I)=HI$(I+1)
2770 HI$(I+1)=Z$
2780 RETURN
2797 :
2798 REM display hi-score table
2799 :
2800 CLS:PAPER 0:INK 7
2810 A$=CHR$(6)+CHR$(10)+"Midnight Fe
ast "+CHR$(5)+"Hi-Scores"
2820 PLOT2,1,A$:PLOT2,2,A$
2830 FOR I=1 TO 8
2840 J=I+(I>6)*5
2850 Z=I*2+3
2860 PLOT7,Z,STR$(I):PLOT7,Z,J

```

```

2870 PLOT12,Z,STR$(HI(I)):PLOT12,Z,J
2880 PLOT10,Z,">":PLOT21,Z,HI$(I)
2890 NEXT I
2895 RETURN
2897 :
2898 REM enter new hi-score
2899 :
2900 PLOT6,22,CHR$(6)+"Please enter y
our name"
2910 FORI=1TO23:PRINT:NEXT I
2920 K$=KEY$: POKE #26A,3 :PRINTCHR$(
20)
2930 INPUT N$
2940 PRINTCHR$(20)CHR$(17)CHR$(30)CHR
$(6)
2950 IF LEN(N$)>16THENN$=LEFT$(N$,16)
2960 HI(8)=S:HI$(8)=N$:SWAP=2750
2970 FOR I=7 TO 1 STEP-1
2980 IF HI(I)<HI(I+1)THEN GOSUB SWAP
2990 NEXT I
2995 GOTO 2800 ' display new table
2997 :
2998 REM user defined graphics
2999 :
3000 FOR I=65 TO 98
3010 :READ D$
3020 :FOR J=1 TO 15 STEP 2
3030 : U=VAL("#"+MID$(D$,J,2))
3040 : P=#B800+I*8+(J-1)/2
3050 : POKE P,U
3060 :NEXT J
3070 NEXT I
3080 RETURN
3090 :
3100 DATA 060F173F0F03060C
3101 DATA 261F0505041C1331
3102 DATA 060F173F0F030E02
3103 DATA 070B13020606050D
3104 DATA 060F1B3F0F030E02
3105 DATA 130F030609091903
3106 DATA 183C3A3F3C30180C
3107 DATA 192E2828080E3223
3108 DATA 183C3A3F3C301C10
3109 DATA 383432101818282C
3110 DATA 183C363F3C301C10
3111 DATA 323C301824242630
3112 DATA 00000029393E0E0A
3113 DATA 00000025271E1C14
3114 DATA 001E0033001E0033
3115 DATA 001E003300000000
3116 DATA 3F3E3C3C38303020
3117 DATA 3F1F0F0F07030301
3118 DATA 0000003F003F0000
3119 DATA 0102020408101020
3120 DATA 2010100804020201
3121 DATA 3F00000000000000

```


MIDNIGHT FEAST

```

3122 DATA 3F20202020202020
3123 DATA 3F01010101010101
3124 DATA 000C1E3F3F3F1E0C
3125 DATA 00002030383C3E00
3126 DATA 000001070A323707
3127 DATA 1008003F3D3D3E3C
3128 DATA 000818303030180C
3129 DATA 00000408183C3C18
3130 DATA 000020202E3F3F3E
3131 DATA 3F2A2A2A3F202020
3132 DATA 3F2A2A2A3F000000
3133 DATA 3E2A2A2A3E020202
3197 :
3198 REM store machine code
3199 :
3200 READ AD : REM works on ANY Oric
3205 ATMOS=(DEEK(FFFFE)=580)
3210 REPEAT
3220 :READ D$
3230 :REPEAT
3240 : V=VAL("#"+D$)
3250 : POKE AD,V
3260 : AD=AD+1 :CHK=CHK+V
3270 : READ D$
3280 :UNTIL D$="OK"
3290 :READ AD
3300 UNTIL AD=FFFF
3305 IF ATMOS THEN GOTO 3320
3310 FOR AD=#4A0 TO #4A3 ' Oric-1
3315 POKE AD,PEEK(AD)+1 ' changes
3317 NEXT AD
3320 IF CHK=9722 THEN RETURN
3328 REM checksum error
3330 CLS:PAPER7:INK0
3340 PRINTCHR$(6)CHR$(17)CHR$(20)
3350 PRINT"Machine code error"CHR$(10)
]
3360 PRINT"Check the data .."
3370 LIST 3400-3413
3380 STOP
3400 DATA#400 : REM move cats
3401 DATA A2,03,BD,A0,04,85,50,BD
3402 DATA A4,04,85,51,BC,A8,04,A9
3403 DATA 20,91,50,BD,AC,04,30,13
3404 DATA C0,17,F0,08,C8,98,9D,A8
3405 DATA 04,4C,3D,04,A9,FF,9D,AC
3406 DATA 04,D0,09,C0,00,F0,09,88
3407 DATA 98,9D,A8,04,A9,4D,D0,07
3408 DATA A9,01,9D,AC,04,A9,4E,91
3409 DATA 50,CA,10,BE,60,OK
3410 DATA#4A0 : REM data table
3411 DATA F3,B8,83,48,BC,BD,BE,BF
3412 DATA 01,07,0F,16,01,FF,01,FF

```

```

3413 DATA OK,FFFF
3497 :
3498 REM play a tune using T$
3499 :
3500 T$=""
3510 T$=T$+"16611665688338&&&"
3520 T$=T$+"6551155113BA86&&&"
3530 TE=3:OC=4:LT=LEN(T$)
3540 FOR I=1 TO LT
3550 :NT=VAL("#"+MID$(T$,I,1)):O=OC
3560 :IF NT>12 THEN O=OC+1:NT=NT-12
3570 :IF NT=0 THEN 3610
3580 :PLAY 0,0,0,0
3590 :MUSIC 1,O,NT,8:MUSIC 2,O-1,NT,4
3600 :PLAY 3,0,1,1000
3610 :WAIT TE
3620 NEXT I
3630 PLAY 0,0,0,0
3640 RETURN
3697 :
3698 REM caught you
3699 :
3700 PAPER 3:FORI=1TO99:NEXT I:PAPER
7:FORI=1TO99:NEXT I:PAPER 0
3710 PLOT 30,26,0:PLOT 0X,OY,NS:PLOT
0X,OY-1,NH:INK 4
3720 M=M-1:PLOT19,26,STR$(M)+".00"
3730 IF M=0 THEN 3800 ' game over
3740 GOSUB 2010 ' reset values
3750 PLOT 30,2,STR$(T)+" "
3755 PULL
3760 GOTO 100 ' restart the main loop
3797 :
3798 REM game over
3799 :
3800 PLOT 15,14,CHR$(12)+"GAME OVER"+
CHR$(9)
3810 WAIT 300:K$=KEY$:PULL
3820 GOSUB 2800 ' display hi-scores
3830 IF S>HI(8) THEN GOSUB 2900
3840 PLOT 6,22,CHR$(3)+"Press <1> for
another game"
3850 PLOT 12,23,CHR$(5)+"<2> to finis
h"
3860 REPEAT:GET K$:UNTIL(K$="1")OR(K$
="2")
3865 GARBAGE=FRE("")
3870 IF K$="1" THEN 50
3880 CLS :PAPER 7:INK 0
3890 PRINTCHR$(6)CHR$(17)CHR$(20);
3900 END
3999 :
4000 REM program by David Reid
4010 REM copyright ' 1984

```

REVENGE OF THE BEANS

LISTING 1

```

5 CLS
10 PRINT"READING DATA NOW..."
15 POKE775,255
20 A=#A000:EL=120:GOSUB500
30 A=#A100:EL=144:GOSUB500
40 A=#A200:EL=182:GOSUB500
50 A=#A300:EL=203
51 GOSUB500
60 A=#A400:EL=236:GOSUB500
70 A=#A500:EL=251:GOSUB500
75 POKE775,39
80 PING:CLS:PRINT" THE CODE IS NOW IN
MEMORY. GET READY"
90 PRINT"A CASSETTE AND PRESS RECORD.
PRESS A"
91 PRINT"KEY WHEN YOU ARE READY TO SA
VE.":GETXX$
92 CSAVE"CODE",A#A000,E#A700:NEW
97 REM -----
98 REM MOVE BEANS AND EXPLOSIONS..
99 REM -----
100 DATA 100 ,78A9BF8501A99085, 1060
101 DATA 101 ,00A217A026B100C9, 761
102 DATA 102 ,61F030C962F045C9, 1194
103 DATA 103 ,711007C96C30034C, 572
104 DATA 104 ,94A0C971F04F3008, 997
105 DATA 105 ,C9791004A9209100, 688
106 DATA 106 ,88D0DA38A500E928, 1056
107 DATA 107 ,8500B002C601CAD0, 920
108 DATA 108 ,CA5860A9209100E0, 956
109 DATA 109 ,17F0E518986928A8, 981
110 DATA 110 ,A96291003898E928, 893
111 DATA 111 ,A84C30A0A9209100, 798
112 DATA 112 ,E017F0CC98186928, 1012
113 DATA 113 ,A8A96191003898E9, 1020
114 DATA 114 ,28A84C30A0A97691, 924
115 DATA 115 ,00C8A97791001898, 809
116 DATA 116 ,6928A8A978910088, 883
117 DATA 117 ,A97791003898E928, 914
118 DATA 118 ,A84C30A0C970D007, 980
119 DATA 119 ,A92091004C30A018, 654
120 DATA 120 ,690191004C30A060, 631
127 REM -----
128 REM MOVE BEAN TIN SIDE TO SIDE.
129 REM -----
130 DATA 130 ,78A404A92099A8BB, 997
131 DATA 131 ,99A9BB99D0BB99D1, 1419
132 DATA 132 ,BB99F8BB99F9BB99, 1517
133 DATA 133 ,20BC9921BCA605E0, 989
134 DATA 134 ,01F00EC88404C023, 818
135 DATA 135 ,D004A20186054C3C, 650
136 DATA 136 ,A1888404C002D004, 839
137 DATA 137 ,A2028605A96399A8, 892
138 DATA 138 ,BB99A9BBA9689920, 1154
139 DATA 139 ,BC9921BCA9E499D0, 1320
140 DATA 140 ,BBA9E599D1BBA9E6, 1533
141 DATA 141 ,99F8BBA9E799F9BB, 1577
142 DATA 142 ,AD1803297FC91410, 605
143 DATA 143 ,06A962EA9949BC58, 1009
144 DATA 144 ,6060555555555555, 702
157 REM -----
158 REM MOVE BASE FROM SIDE TO SIDE
159 REM -----
160 DATA 160 ,A979AC06049990BF, 960
161 DATA 161 ,A9FA99B8BF968BF, 1427
162 DATA 162 ,C920F0034C6FA2B9, 1010
163 DATA 163 ,8FBFC920F0034C6F, 997
164 DATA 164 ,A2B991BFC920F003, 1159
165 DATA 165 ,4C6FA2AD0802C9AC, 905
166 DATA 166 ,F005C9B4F01D60C0, 1183
167 DATA 167 ,03D00160CE0604A9, 693
168 DATA 168 ,209990BF99B8BF88, 1184
169 DATA 169 ,A9799990BFA9FA99, 1350
170 DATA 170 ,B8BF60C025D00160, 1005
171 DATA 171 ,EE0604A9209990BF, 937
172 DATA 172 ,99B8BFC8A9799990, 1315
173 DATA 173 ,BFA9FA99B8BF608D, 1375
174 DATA 174 ,000420CBFAAD0004, 666
175 DATA 175 ,AC0604A9DB9990BF, 1058
176 DATA 176 ,A9DD99B8BFA9018D, 1229
177 DATA 177 ,0104A9148D7602AD, 628
178 DATA 178 ,7602C901D0F9A97B, 1071
179 DATA 179 ,9990BFA97D99B8BF, 1310
180 DATA 180 ,A90F8D7602AD7602, 738
181 DATA 181 ,C901D0F9A9209990, 1157
182 DATA 182 ,BF99B8BF20CBFA60, 1300
197 REM -----
198 REM UPDATE SCORE SCORE AT 48006
199 REM -----
200 DATA 200 ,A400BE85BBE8E03A, 1188
201 DATA 201 ,D00AA2308A9985BB, 1039
202 DATA 202 ,88D0EF608A9985BB, 1290
203 DATA 203 ,6055555555555555, 691
207 REM -----
208 REM MOVE LASER UP AND CHECK HIT
209 REM -----
210 DATA 210 ,A608E001D021AD09, 822
211 DATA 211 ,02C9A5F00160A921, 907
212 DATA 212 ,AC06049990BFA214, 852
213 DATA 213 ,86088409A9BF8507, 783
214 DATA 214 ,A968850620B5FAA6, 1041
215 DATA 215 ,08A409A9209106CA, 735
216 DATA 216 ,8608E001D0016038, 728
217 DATA 217 ,A506E9288506B002, 761
218 DATA 218 ,C607B106C920D005, 834
219 DATA 219 ,A97C910660A20186, 837
220 DATA 220 ,08C961F032C962F0, 1135
221 DATA 221 ,2EC9EBF009C9ECD0, 1376
222 DATA 222 ,04884C66A46038A5, 799
223 DATA 223 ,06E9288506B002C6, 794
224 DATA 224 ,07A971910620CBFA, 925

```

REVENGE OF THE BEANS

```

225 DATA 225 ,A0008D0504A90585, 617
226 DATA 226 ,002000A34C00A3A9, 603
227 DATA 227 ,6D9106A9018DE102, 798
228 DATA 228 ,A9008DE2028DE502, 910
229 DATA 229 ,8DE602A9FF8DE302, 1167
230 DATA 230 ,8DE4022040FBA901, 888
231 DATA 231 ,8DE1028DE502A904, 913
232 DATA 232 ,8DE802A9008DE202, 913
233 DATA 233 ,8DE3028DE4028DE6, 1112
234 DATA 234 ,028DE70220D0FBA9, 1036
235 DATA 235 ,0685002000A34C00, 410
236 DATA 236 ,A360555555555555, 769
237 REM -----
238 REM MOVE SUPER CAN AND PUT BEAN
239 REM -----
240 DATA 240 ,AC0504C000D012AD, 772
241 DATA 241 ,1803297FC9023001, 447
242 DATA 242 ,60A0018C0504209F, 597
243 DATA 243 ,FAAC0504A9209912, 803
244 DATA 244 ,BD9913BD993BBD99, 1104
245 DATA 245 ,3ABDC8C022D006A0, 1047
246 DATA 246 ,008C050460A9EA99, 801
247 DATA 247 ,13BDA9E99912BDA9, 1139
248 DATA 248 ,EC993BBD9A9EB993A, 1252
249 DATA 249 ,BD8C050498184A90, 732
250 DATA 250 ,0160A9629962BD60, 900
251 DATA 251 ,5555555555555555, 680
500 REPEAT:READLL,DA$,CH
501 TT=0
502 FORK=0TO7:B=VAL("#"+MID$(DA$,1+K*
2,2))
503 TT=TT+B:POKEA+K,B
504 NEXT:A=A+8:IFTT<>CH THEN GOTO600
505 UNTIL LL=EL
506 RETURN
600 ZAP:PRINT:PRINT"DATA ERROR FOUND
IN LINE ";LL

```

LISTING 2

```

1 DATA 0, 0, 18, 12, 12, 12, 12, 30
3 DATA 16, 4, 8, 0, 8, 0, 4, 16
7 DATA 12, 30, 43, 63, 45, 33, 30,
12
8 DATA 12, 30, 53, 63, 33, 45, 30,
12
9 DATA 0, 0, 0, 0, 0, 63, 0, 63
10 DATA 63, 63, 33, 32, 36, 36, 36,
33
11 DATA 63, 63, 33, 33, 51, 51, 51,
51
12 DATA 33, 36, 36, 36, 32, 33, 63,
63
13 DATA 51, 51, 51, 51, 51, 51, 63,
63
14 DATA 63, 0, 63, 0, 0, 0, 0, 0
15 DATA 63, 63, 63, 33, 33, 39, 39,
33

```

```

16 DATA 63, 63, 63, 33, 33, 39, 39,
39
17 DATA 33, 57, 57, 33, 33, 63, 63,
63
18 DATA 39, 39, 39, 33, 33, 63, 63,
63
19 DATA 0, 12, 30, 63, 63, 30, 12, 0
20 DATA 0, 0, 12, 30, 30, 12, 0, 0
21 DATA 36, 17, 4, 34, 8, 33, 16, 4
22 DATA 0, 4, 16, 2, 40, 2, 0, 40
23 DATA 56, 48, 34, 4, 8, 1, 1, 7
24 DATA 0, 8, 25, 3, 3, 35, 49, 32
25 DATA 0, 20, 12, 4, 32, 56, 48, 60
26 DATA 0, 0, 12, 12, 28, 1, 6, 6
27 DATA 36, 0, 16, 0, 0, 2, 4, 1
28 DATA 0, 16, 2, 1, 4, 1, 32, 16
29 DATA 0, 16, 4, 8, 0, 0, 16, 4
30 DATA 0, 0, 4, 0, 8, 2, 4, 1
31 DATA 18, 12, 12, 12, 12, 12, 30,
30
32 DATA 63, 45, 45, 33, 45, 45, 63,
33
33 DATA 4, 16, 4, 0, 0, 8, 4, 0
34 DATA 0, 0, 12, 12, 12, 12, 0, 0
35 DATA 0, 4, 34, 16, 0, 0, 8, 0
36 DATA 0, 0, 18, 12, 12, 12, 12, 30
50 REM GRAPHICS DATA PROGRAM
60 REM - -
70 REM REVENGE OF THE BEANS
80 REM - -
90 REM WRITTEN BY DAVID PEAT
100 REM
120 FORN=46808TO46815:READD:POKEN,D:N
EXT
130 FORN=46824TO46831:READD:POKEN,D:N
EXT
140 FORN=46856TO47087:READD:POKEN,D:N
EXT
145 FORN=46344TO46351:READD:POKEN,D:N
EXT
150 ZAP:PRINT"THE GRAPHICS ARE READY.
INSERT YOUR"
160 PRINT"CASSETTE AND PRESS RECORD A
ND PLAY."
170 PRINT"PRESS A KEY WHEN YOU ARE RE
ADY"
180 GETX$:CSAVE"GRAPHICS",A46080,E479
99:PRINT"GRAPHICS ARE NOW SAVED":NEW

```

REVENGE OF THE BEANS

LISTING 3

```

5 POKE618,10:HI$="000000"
10 CLS:T$="SCORE:000000 "+CHR$(17)
)+"REVENGE OF THE BEANS "+CHR$(16)
20 FORN=1TOLN(T$):POKE47999+N,ASC(MI
D$(T$,N,1)):NEXT
30 PAPER4:INK7:FORN=9T010
40 PLOT10,N,CHR$(10)+CHR$(21)+"R E U
E N G E "+CHR$(20)
50 NEXT
60 PLOT11,12,CHR$(18)+"OF THE BEANS
"+CHR$(20)
70 PLOT11,15,CHR$(0)+"BY DAVID PEAT"
73 PLAY2,1,0,0
80 REPEAT
90 RESTORE
100 FORN=1T048:READ OC,NO,TI
110 IFKEY$<>" "THENPING:GOTO200
120 MUSIC1,OC,NO,0:PLAY1,0,1,250*TI
122 PLOT11,12,CHR$(RND(1)*8+16):PLOT1
1,15,CHR$(RND(1)*8)
123 IFTI=8THENTI=10
124 WAITTI*3
125 NEXT:UNTILKEY$<>" "
126 GOTO200
130 DATA3,3,2,3,12,.5,4,1,2,3,10,2,3,
10,2,3,10,8,4,5,2,4,2,.5,4,3,2,4,8,2
131 DATA4,8,2,4,8,8
132 DATA3,12,2,3,9,.5,3,10,2,4,1,2,4,
1,2,4,1,8
133 DATA3,7,2,3,7,.5,3,8,2,3,12,2,3,1
2,2,3,12,8
134 DATA3,3,2,3,12,.5,4,1,2,3,10,2,3,
10,2,3,10,8
135 DATA4,5,2,4,2,.5,4,3,2,4,12,2,4,1
2,2,4,12,8
136 DATA3,12,2,3,12,.5,4,1,2,4,5,2,4,
5,2,4,5,8,3,7,2,3,10,.5,3,8,2,3,8,2
137 DATA3,8,2,3,8,8
200 CLS:PAPER0:INK3
201 FORN=9T010
202 PLOT10,N,CHR$(10)+CHR$(21)+"R E U
E N G E "+CHR$(16)
203 NEXT
204 PLOT11,11,CHR$(20)+"OF THE BEANS
"+CHR$(16)
205 FORN=9T011:PLOT9,N,0:NEXT
206 PLOT11,13,"1.PLAY REVENGE."
207 PLOT11,15,"2.DEMO MODE..."
208 PLOT11,17,"3.INSTRUCTIONS."
209 PLOT14,19,"PRESS KEY"
210 REPEAT:PLOT3,13,RND(1)*7+1:PLOT3,
15,RND(1)*7+1:PLOT3,17,RND(1)*7+1
212 K$=KEY$:UNTILK$<>" "
213 A=VAL(K$):IFA<>1ANDA<>2ANDA<>3THE
N210
215 ON A GOTO 1000,2000,3000

```

```

216 GOTO210
1000 SC$="":FORN=48006T048011:POKEN,4
8:NEXT
1010 CLS:PAPER3:INK5:PLOT1,26,0
1020 POKE4,4:POKE5,1:POKE8,0:POKE1029
,0:POKE1030,20
1030 POKE1025,0
1040 PLOT11,10,CHR$(12)+CHR$(4)+"GET
READY":WAIT100
1050 PLOT11,10,
1060 REPEAT
1070 CALL#A000:CALL#A100:CALL#A400:CA
LL#A200:CALL#A400:CALL#A500:CALL#A400
1080 UNTILPEEK(1025)=1
1090 FORN=48006T048011:SC$=SC$+CHR$(P
EEK(N)):NEXT
1100 CLS:FORN=5T06:PLOT10,N,CHR$(10)+
"YOUR SCORE IS "+SC$:NEXT
1110 FORN=9T010:PLOT6,N,CHR$(10)+"THE
HIGH SCORE IS "+HI$:NEXT
1120 IFVAL(SC$)<VAL(HI$)THEN1200
1130 FORN=13T014:PLOT6,N,CHR$(14)+"YO
U HAVE THE HIGH SCORE":NEXT
1140 HI$=SC$
1200 PLOT13,20,"PRESS A KEY"
1210 GETXX$
1211 GETXX$
1220 GOTO200
2000 FORN=48006T048011:POKEN,48:NEXT
2010 CLS:PAPER1:INK3:PLOT1,26,0
2020 POKE4,4:POKE5,1:POKE8,0:POKE1029
,0:POKE1030,20
2030 POKE1025,0
2040 PLOT10,10,"PRESS ESC TO FOR MENU
":WAIT300
2050 PLOT10,10,"
"
2060 REPEAT
2061 CALL#A000:CALL#A100:CALL#A500
2062 IFRND(1)>.7THENPOKE521,165
2063 IFRND(1)>.6THENPOKE520,180
2064 IFRND(1)<.4THENPOKE520,172
2065 CALL#A200:CALL#A400:CALL#A400:CA
LL#A400
2080 UNTILPEEK(1025)=1ORKEY$=CHR$(22)
2090 FORN=48006T048011:POKEN,48:NEXT:
GOTO200
2999 END
3000 CLS:PAPER5:INK0:PRINT:PRINT" THE
BEANS HAVE DEvised A HORRIFIC"
3010 PRINT:PRINT"REVENGE FOR THEIR YE
ARS OF SUFFERING"
3020 PRINT:PRINT"AT THE HANDS OF THE
HUMAN RACE."
3030 PRINT:PRINT"THEY HAVE COME TO EA

```


REVENGE OF THE BEANS

```
RTH IN GIANT BEAN":PRINT
3040 PRINT"CANS AND CRASH TO THE GROU
ND IN A"
3050 PRINT:PRINT"KAMIKAZE STYLE."
3060 PRINT:PRINT" YOU MUST CONTROL YO
UR ANTI-BEAN RIOT"
3070 PRINT:PRINT"PHASOR LEFT AND RIGH
T TO SPLAT THE"
```

```
3080 PRINT:PRINT"BEANS BEFORE THEY SP
LAT YOU."
3090 PRINT:PRINT:PRINT" MAY THE W
IND BE WITH YOU.. "
3100 PLOT0,26,CHR$(20)+CHR$(3)+""
PRESS RETURN TO GO TO MENU"
3200 REPEAT:UNTILKEY$=CHR$(13)
3300 GOTO200
```

CHRISTMAS CAROLS

```
5 POKE 618,10
10 GRAB
20 DIMA(61,3),B(61),C(19,3),D(19),C1(
19,3),D1(19),E(28,3),F(28),G(28,3),H(2
8)
21 DIMI(19,3),J(19),I1(16,3),J1(16)
25 GOSUB500
30 GOSUB1000
35 REM*****
40 PRINT" 1. O Come all ye faithful
."
45 PRINT
50 PRINT" 2. O little town of Bethl
ehem."
55 PRINT
60 PRINT" 3. The Coventry Carol."
62 PRINT
64 PRINT" 4. While shepherds watche
d"
65 PRINT
66 PRINT" 5. Once in royal David's
City."
67 PRINT:PRINT
70 PRINT" Press a number to choose"
80 GET CH$:CH=VAL(CH$):ON CH GOSUB200
0,2500,3000,3500,4000
90 GOTO80
100 REM*****
500 TEXT:CLS:PAPER1:INK3
510 PRINT:PRINT:PRINT
520 TITLE$=CHR$(138)+"CAROLS "
530 PRINTSPC(10)TITLE$
540 PRINTSPC(10)TITLE$
550 PRINT:PRINT:PRINT
560 RETURN
1000 REM***** O COME
1010 FORX=1TO61
1020 FORY=1TO3
1030 READA(X,Y):NEXT
1040 READ B(X):NEXT
1050 REM***** O LITTLE
1055 FORQ=1 TO 19
1060 FORW=1 TO 3
1065 READC(Q,W):NEXT
1070 READD(Q):NEXT
1075 FORQ1=1 TO 19
1080 FORW1=1 TO 3
1085 READC1(Q1,W1):NEXT
```

```
1090 READD1(Q1):NEXT
1100 REM***** COVENTRY
1110 FORO=1 TO 28
1120 FORP=1 TO 3
1130 READE(O,P):NEXT
1140 READF(O):NEXT
1150 REM***** SHEPHERDS
1200 FORM= 1 TO 28
1210 FORN=1 TO 3
1220 READG(M,N):NEXT
1230 READH(M):NEXT
1300 REM*****DAVID
1320 FORK=1 TO 19
1330 FORL=1 TO 3
1340 READI(K,L):NEXT
1350 READJ(K):NEXT
1370 FORK1=1 TO 16
1380 FORL1= 1 TO 3
1390 READI1(K1,L1):NEXT
1400 READJ1(K1):NEXT
1410 RETURN
1420 REM*****PLAYS TUNES
1999 REM O COME ALL YE FAITHFUL
2000 FORX=1 TO 61
2010 FORY=1 TO 3
2020 SOUND1,A(X,1),0:SOUND2,A(X,2),0:
SOUND3,A(X,3),0
2030 PLAY7,0,1,5000:NEXT
2040 WAIT B(X)*40:NEXT
2050 RETURN
2060 REM*****
2499 REM O LITTLE TOWN
2500 REPEAT:CO=CO+1
2505 FOR Q=1 TO 19
2510 FOR W=1 TO 3
2520 SOUND1,C(Q,1),0:SOUND2,C(Q,2),0:
SOUND3,C(Q,3),0
2530 PLAY7,0,1,5000:NEXT
2540 WAIT D(Q)*40:NEXT
2545 UNTIL CO=2
2600 FOR Q1=1 TO 19
2610 FOR W1=1 TO 3
2620 SOUND1,C1(Q1, 1),0:SOUND2,C1(Q1,
2),0:SOUND3,C1(Q1,3),0
2630 PLAY7,0,1,5000:NEXT
2640 WAITD1(Q1)*40:NEXT
2650 FOR Q=1 TO 19:REM REPEAT LINE
2660 FOR W=1 TO 3
```

CHRISTMAS CAROLS

```

2670 SOUND1,C(Q,1),0:SOUND2,C(Q,2),0:
SOUND3,C(Q,3),0
2680 PLAY7,0,1,5000:NEXT
2690 WAIT D(Q)*40:NEXT
2700 RETURN
2710 REM*****
2999 REM COVENTRY CAROL
3000 FOR O=1 TO 28
3010 FOR P=1 TO 3
3020 SOUND1,E(O,1),0:SOUND2,E(O,2),0:
SOUND3,E(O,3),0
3030 PLAY7,0,1,5000:NEXT
3040 WAIT F(O)*50:NEXT
3050 RETURN
3060 REM*****
3500 REM WHILE SHEPHERDS
3510 FOR M=1 TO 28
3520 FOR N=1 TO 3
3530 SOUND1,G(M,1),0:SOUND2,G(M,2),0:
SOUND3,G(M,3),0
3540 PLAY7,0,1,5000:NEXT
3550 WAIT H(M)*40:NEXT
3560 RETURN
3570 REM*****
3999 REM DAVID
4000 REPEAT:DO=DO+1
4010 FORK=1 TO 19
4020 FORL=1 TO 3
4030 SOUND1,I(K,1),0:SOUND2,I(K,2),0:
SOUND3,I(K,3),0
4040 PLAY7,0,1,5000:NEXT
4050 WAIT J(K)*50:NEXT
4060 UNTIL DO=2
4070 FORK1=1 TO 16
4080 FORL1=1 TO 3
4090 SOUND1,I1(K1,1),0:SOUND2,I1(K1,2
),0:SOUND3,I1(K1,3),0
4100 PLAY7,0,1,5000:NEXT
4110 WAIT J1(K1)*50:NEXT
4120 RETURN
5000 REM*****
5070 DATA 158,252,317,1
5080 DATA 158,252,317,2
5090 DATA 211,252,317,1
5100 DATA 158,252,317,1
5110 DATA 141,237,317,2
5120 DATA 211,283,336,2
5130 DATA 125,211,317,1
5140 DATA 141,211,316,1
5150 DATA 125,211,317,1
5160 DATA 118,188,476,1
5170 DATA 125,158,424,2
5180 DATA 141,167,424,1
5190 REM
5200 DATA 158,252,378,1
5210 DATA 158,252,378,2

```

```

5220 DATA 167,211,424,1
5230 DATA 188,224,283,1
5240 DATA 167,283,424,1
5250 DATA 158,211,252,1
5260 DATA 141,211,336,1
5270 DATA 125,211,317,1
5280 DATA 167,211,283,2
5290 DATA 188,224,283,1.5
5300 DATA 211,283,424,0.5
5310 DATA 211,336,424,4
5320 REM
5330 REM
5340 DATA 105,211,252,2
5350 DATA 118,188,283,1
5360 DATA 125,211,317,1
5370 DATA 118,158,283,2
5380 DATA 125,211,317,2
5390 DATA 141,211,336,1
5400 DATA 125,211,317,1
5410 DATA 158,252,378,1
5420 DATA 141,188,476,1
5430 DATA 167,283,424,2
5440 DATA 211,336,424,1
5450 REM
5460 REM
5470 DATA 158,252,317,1
5480 DATA 158,252,317,1
5490 DATA 167,283,424,1
5500 DATA 158,252,378,1
5510 DATA 141,237,424,1
5520 DATA 158,252,317,2
5530 DATA 211,252,424,1
5540 DATA 125,211,317,1
5550 DATA 125,211,317,1
5560 DATA 141,237,336,1
5570 DATA 125,211,317,1
5580 DATA 118,188,283,1
5590 DATA 125,211,317,2
5600 DATA 141,167,424,1
5610 REM
5620 REM
5630 DATA 125,211,317,1
5640 DATA 118,188,283,1
5650 DATA 125,211,317,1
5660 DATA 141,211,336,1
5670 DATA 158,224,378,1
5680 DATA 167,283,424,2
5690 DATA 158,211,252,1
5700 DATA 118,188,476,1
5710 DATA 125,158,424,2
5720 DATA 141,167,424,1.5
5730 DATA 158,252,317,.4
5740 DATA 158,252,317,4
5750 REM*****
6220 REM
6230 DATA 237,317,378,1

```

CHRISTMAS CAROLS

6240 DATA 177,283,476,1
 6250 DATA 177,211,267,1
 6260 DATA 177,237,283,1
 6270 DATA 158,267,317,1
 6280 DATA 141,237,356,.3
 6290 DATA 158,237,356,.3
 6300 DATA 141,177,424,.3
 6310 DATA 133,177,424,.3
 6320 DATA 118,188,476,1
 6330 DATA 141,177,424,1
 6340 DATA 133,211,317,1
 6350 DATA 141,237,283,.3
 6360 DATA 177,237,283,.3
 6370 DATA 158,211,267,1
 6380 DATA 158,188,476,1
 6390 DATA 177,283,356,1
 6400 DATA 177,267,356,1
 6410 DATA 177,283,356,1
 6420 REM
 6430 REM
 6440 DATA 177,283,424,.3
 6450 DATA 141,283,424,.3
 6460 DATA 118,188,283,1.5
 6470 DATA 105,177,267,.3
 6480 DATA 118,188,237,.35
 6490 DATA 133,188,237,.3
 6500 DATA 141,237,424,.3
 6510 DATA 158,267,378,.3
 6520 DATA 177,283,356,.3
 6530 DATA 158,283,356,.3
 6540 DATA 141,177,211,.3
 6550 DATA 133,177,211,.3
 6560 DATA 118,188,476,1
 6570 DATA 237,378,476,1
 6580 DATA 177,237,283,1
 6590 DATA 141,237,356,1
 6600 DATA 158,267,317,1
 6610 DATA 177,211,267,1
 6620 DATA 237,378,476,3
 6630 REM*****
 7000 DATA 159,212,318,1
 7010 DATA 159,212,318,1
 7020 DATA 168,212,425,1
 7030 DATA 159,212,637,2
 7040 REM
 7050 DATA134,212,535,1
 7060 DATA142,178,357,2
 7070 DATA159,268,401,1
 7080 DATA168,212,425,3
 7090 REM
 7100 DATA159,212,318,1
 7110 DATA142,178,357,1
 7120 DATA134,212,535,1
 7130 DATA119,200,477,1
 7140 DATA142,212,425,2
 7150 DATA159,253,318,2

7160 REM
 7170 DATA106,178,535,1
 7180 DATA119,178,357,2
 7190 DATA134,212,535,1
 7200 DATA142,212,425,2
 7210 REM
 7220 DATA134,212,535,1
 7230 DATA142,178,357,2
 7240 DATA159,268,401,1
 7250 DATA168,212,425,3
 7260 REM
 7270 DATA159,212,318,1
 7280 DATA168,212,425,1
 7290 DATA159,212,318,1
 7300 DATA119,200,477,1
 7310 DATA142,212,425,2
 7320 DATA126,212,318,3
 8000 DATA178,238,357,1
 8010 DATA142,238,357,1.5
 8020 DATA142,238,357,.5
 8030 DATA159,189,477,1
 8040 DATA178,212,425,1
 8050 DATA134,212,535,1
 8060 DATA 134,178,535,1
 8070 DATA142,178,357,1
 8080 REM
 8090 DATA159,189,477,1
 8100 DATA142,178,357,1
 8110 DATA119,189,284,1
 8120 DATA119,212,318,1
 8130 DATA126,212,318,1
 8140 DATA119,189,477,3
 8150 REM
 8160 DATA142,238,357,1
 8170 DATA106,178,268,1.5
 8180 DATA119,178,357,.5
 8190 DATA134,178,425,1
 8200 DATA142,238,357,1
 8210 DATA159,189,477,1
 8220 DATA178,284,425,1
 8230 DATA189,224,569,1
 8240 REM
 8250 DATA142,238,357,1
 8260 DATA159,189,477,1
 8270 DATA178,284,425,1
 8280 DATA178,212,535,1
 8290 DATA189,318,477,1
 8300 DATA178,284,715,3
 9000 DATA212,253,318,1
 9010 DATA168,238,284,1
 9020 DATA159,212,253,1.5
 9030 DATA159,253,318,.3
 9040 DATA159,253,425,.3
 9050 DATA168,284,425,.3
 9060 DATA159,253,425,.3
 9070 DATA142,238,425,.3

CHRISTMAS CAROLS

```

9080 DATA142,238,425,1
9090 DATA159,253,318,1
9100 REM
9120 DATA159,253,318,1
9130 DATA125,159,379,1
9140 DATA106,253,318,1.5
9150 DATA126,212,318,.3
9160 DATA126,189,477,.3
9170 DATA142,238,477,.3
9180 DATA159,253,425,.3
9190 DATA168,284,425,.3
9200 DATA159,253,318,2
9210 REM
9220 DATA94,159,238,1
9230 DATA94,159,238,1

```

```

9240 DATA106,253,318,1.5
9250 DATA159,253,318,.5
9260 DATA119,159,425,1
9270 DATA119,168,425,1
9280 DATA126,212,318,2
9290 REM
9300 DATA94,159,238,1
9310 DATA94,159,238,1
9320 DATA106,253,318,1.5
9330 DATA126,212,318,.3
9340 DATA126,212,477,.3
9350 DATA142,238,477,.3
9360 DATA159,253,425,.3
9370 DATA168,284,425,.3
9380 DATA159,253,637,2

```

ATMOS BASIC UTILITIES

LISTING 1

LISTING 2

```

10 'CLOCKBBASIC
20 'DAVID SINFIELD
30 '(0485) 70865
40 '
100 FORN=#9400TO#9465
110 READP
120 POKEN,P
130 NEXT
1000 DATA#A9,#10,#8D,#80,#BB,#A9,#07,
#8D,#81,#BB,#A9,#30,#A0,#0B
1010 DATA#99,#82,#BB,#88,#F0,#17,#C0,
#09,#F0,#0A,#C0,#06,#F0,#06
1020 DATA#C0,#03,#F0,#02,#D0,#EC,#A9,
#3A,#99,#82,#BB,#A9,#30,#D0
1030 DATA#E6,#60,#78,#48,#98,#48,#8A,
#48,#A0,#0B,#B9,#82,#BB,#C9
1040 DATA#3A,#F0,#19,#C0,#04,#F0,#1A,
#C0,#07,#F0,#16,#C9,#39,#F0
1050 DATA#08,#18,#69,#01,#99,#82,#BB,
#D0,#10,#A9,#30,#99,#82,#BB
1060 DATA#88,#F0,#08,#D0,#DB,#C9,#35,
#F0,#F2,#D0,#E8,#68,#AA,#68
1070 DATA#A8,#68,#58,#40

```

```

10 ' AUTO LINE NUMBER
20 ' DAVID SINFIELD
30 ' (0485) 70865
40 '
90 HIMEM#954B
100 PRINT"COPYING AND AMENDING ROM RO
UTINE"
110 P=#9674
115 FORN=#C4D3TO#C55C
120 POKEP,PEEK(N)
130 P=P+1
140 NEXT
150 DOKE#96FE,#9561
155 POKE#96CE,#2E
160 PRINT:PRINT:PRINT"READING & POKIN
G ORIGINAL BIT
170 FORN=#954CTO#9673
180 READP
185 CS=CS+P
190 POKEN,P
200 NEXT
202 IF CS<>31577 THENPRINT"CHEKSUM ER
ROR":END

```


ATMOS BASIC UTILITIES

```

210 DOKE#2F5,#954C
10000 DATA#20,#6A,#95,#20,#7F,#95,#20
,#E1,#95,#20,#2F,#C8,#20,#0C
10010 DATA#96,#20,#94,#C5,#4C,#5F,#96
,#20,#2D,#96,#4C,#55,#95,#4C
10020 DATA#C4,#95,#A2,#03,#A9,#00,#9D
,#05,#04,#CA,#10,#FA,#A2,#04
10030 DATA#A9,#30,#9D,#00,#04,#CA,#10
,#FA,#60,#A9,#00,#85,#EA,#A9
10040 DATA#35,#85,#E9,#A2,#06,#A9,#2C
,#85,#00,#20,#A2,#95,#E0,#05
10050 DATA#F0,#23,#A2,#04,#A9,#00,#85
,#00,#20,#A2,#95,#E0,#03,#F0
10060 DATA#16,#60,#20,#E2,#00,#CA,#30
,#0F,#C5,#00,#F0,#0A,#C9,#30
10070 DATA#30,#07,#C9,#3A,#10,#03,#30
,#EC,#60,#A0,#00,#BE,#CA,#95
10080 DATA#F0,#06,#20,#7C,#F7,#C8,#D0
,#F5,#A2,#FF,#9A,#4C,#B0,#C4
10090 DATA#41,#55,#54,#4F,#20,#4C,#49
,#4E,#45,#20,#4E,#55,#4D,#42
10100 DATA#45,#52,#20,#45,#52,#52,#4F
,#52,#00,#A0,#08,#A2,#00,#C6
10110 DATA#E9,#A1,#E9,#C9,#2C,#F0,#08
,#29,#0F,#99,#00,#04,#88,#D0
10120 DATA#F0,#A0,#05,#A9,#00,#99,#00
,#04,#88,#C6,#E9,#A1,#E9,#C9
10130 DATA#2F,#30,#06,#99,#00,#04,#4C
,#FC,#95,#60,#A0,#00,#B9,#00
10140 DATA#04,#C8,#C9,#30,#F0,#F8,#A2
,#00,#88,#B9,#00,#04,#F0,#0E
10150 DATA#86,#00,#AA,#20,#7C,#F7,#A6
,#00,#95,#35,#E8,#C8,#D0,#ED
10160 DATA#60,#A2,#02,#A0,#04,#B9,#00
,#04,#18,#7D,#06,#04,#99,#00
10170 DATA#04,#C9,#3A,#30,#1B,#98,#48
,#38,#B9,#00,#04,#E9,#0A,#99
10180 DATA#00,#04,#88,#18,#B9,#00,#04
,#69,#01,#99,#00,#04,#C9,#3A
10190 DATA#10,#E9,#68,#A8,#88,#CA,#10
,#D3,#60,#A2,#34,#A0,#00,#86
10200 DATA#E9,#84,#EA,#20,#E2,#00,#AA
,#D0,#03,#4C,#C4,#95,#A2,#FF
10210 DATA#86,#A9

```

LISTING 3

```

10 ' CREATE DATA FROM MEMORY
20 ' DAVID SINFIELD
30 ' (0485) 70865
40 '
100 HIMEM#3FFF
110 PRINT"COPYING AND AMENDING ROM RO
UTINE"

```

```

115 P=#4151
120 FORN=#C4D3 TO#C55C
130 POKEP,PEEK(N)
140 P=P+1
150 NEXT
160 DOKE#41DB,#4027
170 POKE#41AB,#2E
180 PRINT:PRINT:PRINT"READING AND POK
ING ORIGINAL BIT"
190 FORN=#4000 TO #4150
200 READP
210 POKEN,P
215 CS=CS+P
220 NEXT
230 IF CS<>33254 THENPRINT"CHECKSUM E
RROR":END
300 CLS
310 PRINT@1,10;:PRINT
320 INPUT "START OF BLOCK ";S
330 INPUT "END OF BLOCK ";E
335 IFE-S<1THENPRINT"TRY AGAIN":GOTO3
10
340 PRINT:PRINT
350 INPUT"START LINE NUMBER ";SL
360 INPUT"INCREMENT ";IN
370 FL=(E-S)/14*IN+SL
380 IFFL>63999 THEN PRINT"THIS WILL C
AUSE AN ILLEGAL LINE NO.":GOTO320
390 A$=STR$(SL)
395 A=#404F
400 FORN=LEN(A$)TO2STEP-1
410 POKEA,ASC(MID$(A$,N,1))
420 A=A-1
430 NEXT
490 A$=STR$(IN)
495 A=#4053
500 FORN=LEN(A$)TO2STEP-1
510 POKEA,VAL(MID$(A$,N,1))
520 A=A-1
530 NEXT
540 PRINT"IF YOU HAVE NOT SAVED THE P
ROGRAM"
550 PRINT"DO IT NOW.
560 PRINT"PRESS RETURN TO CONINJE PRO
GRAM"
570 PRINT"ANY OTHER KEY TO HALT"
580 A$=KEY$:GETA$
590 IF ASC(A$)<>13 THEN END
600 DOKE#4062,S
610 DOKE#4064,E+1
620 PRINT"TYPE CALL#4000 TO PRODUCE D
ATA"
630 NEW
10000 DATA#20,#3E,#40,#20,#2F,#C8,#20
,#E9,#40,#20,#66,#40,#A0,#00
10010 DATA#B1,#00,#20,#81,#40,#20,#C3

```

ATMOS BASIC UTILITIES

```
,#40,#E8,#A5,#05,#30,#04,#E0
10020 DATA#40,#30,#ED,#CA,#A9,#00,#95
,#35,#4C,#3C,#41,#A5,#05,#30
10030 DATA#10,#20,#0A,#41,#A9,#0D,#20
,#D9,#CC,#A9,#0A,#20,#D9,#CC
10040 DATA#4C,#03,#40,#4C,#E3,#40,#A0
,#20,#B9,#4B,#40,#99,#00,#04
10050 DATA#30,#0C,#C8,#D0,#F5,#30,#30
,#30,#30,#30,#00,#00,#00,#00
10060 DATA#A0,#00,#B9,#62,#40,#99,#00
,#20,#C8,#C0,#04,#D0,#F5,#60
10070 DATA#00,#40,#51,#41,#A0,#00,#B9
,#7C,#40,#F0,#0E,#86,#05,#AA
10080 DATA#20,#7C,#F7,#A6,#05,#95,#35
,#C8,#E8,#D0,#ED,#60,#44,#41
10090 DATA#54,#41,#00,#48,#48,#A9,#23
,#20,#B9,#40,#95,#35,#E8,#68
10100 DATA#4A,#4A,#4A,#4A,#09,#30,#20
,#B1,#40,#20,#B9,#40,#95,#35
10110 DATA#E8,#68,#29,#0F,#09,#30,#20
,#B1,#40,#20,#B9,#40,#95,#35
10120 DATA#E8,#A9,#2C,#95,#35,#20,#B9
,#40,#60,#C9,#3A,#30,#03,#18
```

```
10130 DATA#69,#07,#60,#86,#05,#AA,#20
,#7C,#F7,#8A,#A6,#05,#60,#A9
10140 DATA#00,#85,#05,#18,#A5,#00,#69
,#01,#85,#00,#A5,#01,#69,#00
10150 DATA#85,#01,#C5,#03,#D0,#0A,#A5
,#00,#C5,#02,#D0,#04,#A9,#FF
10160 DATA#85,#05,#60,#A2,#FF,#9A,#4C
,#80,#C4,#A0,#00,#B9,#00,#04
10170 DATA#C8,#C9,#30,#F0,#F8,#A2,#00
,#88,#B9,#00,#04,#F0,#0E,#86
10180 DATA#05,#AA,#20,#7C,#F7,#A6,#05
,#95,#35,#E8,#C8,#D0,#ED,#60
10190 DATA#A2,#02,#A0,#04,#B9,#00,#04
,#18,#7D,#06,#04,#99,#00,#04
10200 DATA#C9,#3A,#30,#1B,#98,#48,#38
,#B9,#00,#04,#E9,#0A,#99,#00
10210 DATA#04,#88,#18,#B9,#00,#04,#69
,#01,#99,#00,#04,#C9,#3A,#10
10220 DATA#E9,#68,#A8,#88,#CA,#10,#D3
,#60,#A2,#34,#A0,#00,#86,#E9
10230 DATA#84,#EA,#20,#E2,#00,#AA,#D0
,#03,#4C,#E3,#40,#A2,#FF,#86
10240 DATA#A9
```

CRIBBAGE

```
1 REM
2 REM
3 REM
6 REM *****
7 REM *      CRIBBAGE  * ===== *
8 REM *      DEC 1983  *'kernow'*
9 REM *      ' BY L.ROOKE * ===== *
10 REM *****
15 U9#=1
20 IFU9#=0THENGOSUB6500
25 CLS:PAPER3:INK0
30 TI$="PLEASE WAIT":CU=11:GOSUB800
40 GOSUB7000:IFU9#=0THEN60
43 PLOT1,15,12:PLOT2,15,4:PLOT6,15,"Pre
ss ANY KEY to continue"
45 REPEAT:PING:WAIT50:X=RND(1):UNTILKEY
$<>" "
47 PRINTCHR$(17);CHR$(6)
50 GOSUB10000
60 PAPER6:INK0:CLS
70 TI$="Do you want to read INSTRUCTION
S"
75 CU=12:GOSUB800:GOSUB480
80 IF A$="Y" THEN GOSUB12000
85 PAPER 2:INK 4:GOSUB9000
95 CLS:TI$=""
100 TI$=UU$+" PLEASE "+TI$+"CUT FOR DEA
L":CU=8:GOSUB800
```

```
110 TI$="Press any key to cut CARDS"
120 CU=12:GOSUB 800
130 GET A$
135 CT(P)=INT(RND(1)*52)+1:IFVAL(CA$(CT
(P)))<1 THEN 135
140 B1$=UU$+" YOUR CARD IS "
145 PLOT(38-LEN(B1$))/2,14,B1$:Q9=P:GOS
UB430
150 WAIT 100
155 CT(O)=INT(RND(1)*52)+1:IFVAL(CA$(CT
(O)))<1 THEN 155
160 B2$="ORIC CARD IS "
165 PLOT(38-LEN(B2$))/2,20,B2$:Q9=0:GOS
UB430
170 CP=VAL(CA$(CT(P))):CO=VAL(CA$(CT(O)
))
175 IFCP<>COTHEN190
180 PLOT2,26,12:PLOT4,26,7
185 B3$=" CARDS ARE EQUAL":PLOT8,26
,B3$:WAIT 400:CLS:TI$="RE-":GOTO100
190 IF CP>CO THEN LEAD=TRUE:DEAL=FALSE
200 IF CP<CO THEN LEAD=FALSE:DEAL=TRUE
210 WAIT 200
220 GOSUB1000
300 REM
301 REM
302 REM
303 REM*****
```

CRIBBAGE

```

304 REM* *
305 REM*      MAIN  LOOP      *
306 REM* *
307 REM*****
308 REM
309 REM
310 REPEAT
320 GOSUB 1200 :REM SHUFFLE
330 GOSUB 1400 :REM DEAL
340 GOSUB 1800 :REM FIND ORIC HAND
350 GOSUB 2200 :REM PUT CARD IN BOX
360 GOSUB 2400 :REM CUT 5th CARD
370 GOSUB 2600 :REM PLAY CARDS
373 GOSUB 8100 :REM PL$(X)=PL$(X+6)
380 IF DEAL=FALSE THEN 387
383 PF#=P:GOSUB4500
385 PF#=0:GOSUB4500:GOSUB5300:GOTO393
387 PF#=0:GOSUB4500
390 PF#=P:GOSUB4500:GOSUB5300
391 REM CHANG LEAD
393 IF DEAL=FALSE THEN DEAL=TRUE:LEAD=F
ELSE:GOTO398
396 IF DEAL=TRUE THEN DEAL=FALSE:LEAD=T
RUE
398 GOSUB8000
400 UNTIL SC(P)=>121 OR SC(O)=>121
410 GOSUB 6000 :REM WINNER + NEW GAME
420 END REM*****
430 IFQ9=0THENQ8=22ELSEQ8=16
435 FORJJ=Q8TOQ8+2:PLOT20,JJ,18:PLOT17,
JJ,23
440 PLOT16,JJ,ASC(MID$(CA$(CT(Q9)),3,1)
)
445 NEXTJJ
450 PLOT18,Q8,RIGHT$(CA$(CT(Q9)),2)
455 PLOT18,Q8+1,MID$(CA$(CT(Q9)),4,2)
460 PLOT18,Q8+2,RIGHT$(CA$(CT(Q9)),2)
465 RETURN
479 REM *** YES/NO ***
480 TI$="Please press 'Y' or 'N'"
490 CU=CU+2:IFCU>24THENCU=2:CLS
495 GOSUB 800:A$=""
500 REPEAT:A$=KEY$:X=RND(9):UNTILA$<>" "
510 IFA$<>"Y"AND A$<>"N"THEN480
520 RETURN
800 IFCU>25THENCU=25
805 PLOT2,CU,SP$
810 PLOT2,CU+1,SP$
820 TI=INT(((38-LEN(TI$))/2)
830 IFTI<2THENTI=2
840 PLOTTI,CU,TI$
850 RETURN
999 REM ***SET UP PEG BORD.***
1000 CLS:PAPER2 :INK0
1010 FORJ=48080TO48280STEP40:POKEJ,23:N
EXTJ

```

```

1020 PRINTSPC(37-LEN(UU$))IC$(1)UU$:A$=
"ORIC":PLOT34,7,4:PLOT35,7,A$
1040 FORJ=2TO5STEP3:FORK=1TO32STEP6:FOR
L=1TO5:PLOTK+L,J,"!":NEXTL,K
1050 PLOT37,J,"":NEXTJ
1060 PLOT 37,1,"J":PLOT 37,6,"J"
1070 PLOT 1,1,1:PLOT 1,3,1
1080 PLOT 1,4,4:PLOT 1,6,4
1090 RETURN
1199 REM *** SHUFFLE ***
1200 TI$="SHUFFLING Please wait":CU=25
:GOSUB 800
1210 FOR J=1 TO INT(RND(1)*20)+80
1220 S0=INT(RND(1)*52)+1:S1=INT(RND(1)*
52)+1
1230 R$=CA$(S0):CA$(S0)=CA$(S1):CA$(S1)
=R$
1240 NEXT J
1250 NC=1:RETURN
1399 REM *** DEAL ***
1400 FOR C=1 TO 6
1403 IF VAL(CA$(NC))<1 THEN NC=NC+1:GOT
O1403
1405 HND$(C,O)=CA$(NC):NC=NC+1
1410 IF VAL(CA$(NC))<1 THEN NC=NC+1:GOT
O1410
1415 HND$(C,P)=CA$(NC):NC=NC+1
1420 NEXT C
1425 TI$="YOUR DEAL "+UU$:IFLEAD=FALSE
HENTI$="ORIC to DEAL"
1427 CU=25:GOSUB800
1430 TI$="YOUR BOX "+UU$:IFLEAD=FALSE
HENTI$="ORIC'S BOX"
1435 PLOT3,7,S4$:CU=7:GOSUB820:WAIT100
1440 C=1: FOR CE=5 TO 30 STEP 5
1449 REM*ORIC LEAD
1450 IF DEAL=FALSETHEN GOSUB1600: GOSU
B1650
1459 REM *PLAYER LEAD
1470 IF DEAL=TRUE THEN GOSUB1650:GOSUB1
600
1490 NEXT CE
1500 PLOT 35,12,"BOX"
1510 FORJJ=13TO15
1520 PLOT34,JJ,3
1530 PLOT1,JJ,3
1540 PLOT35,JJ,"@@@"
1550 PLOT2,JJ,"@@@"
1560 NEXTJJ
1595 RETURN
1599 REM *PLOT ORIC CARD*
1600 FORJJ=9TO11
1610 PLOTCE,JJ,3
1620 PLOTCE+1,JJ,"@@@"
1630 NEXTJJ
1645 WAIT 50:RETURN

```

CRIBBAGE

```

1649 REM *POLT PLAYER CARD*
1650 FORJJ=20TO22:PLOTCE+4,JJ,18:PLOTCE
+1,JJ,23
1660 PLOTCE,JJ,ASC(MID$(HND$(C,P),3,1))
1670 NEXTJJ:JJ=0
1700 PLOTCE+1,18,0
1710 PLOT CE+2,18,48+C
1720 PLOT CE+2,20,RIGHT$(HND$(C,P),2)
1730 PLOT CE+2,21,MID$(HND$(C,P),4,2)
1740 PLOT CE+2,22,RIGHT$(HND$(C,P),2)
1750 WAIT 50:C=C+1:RETURN
1799 REM *** FIND ORIC HAND ***
1800 TI$="Please wait ORIC IS THINKING"
:CU=25:GOSUB800
1804 DT=15
1805 FORJ1=1 TO LV:CC(J1)=INT(RND(1)*DT
)+1
1810 FOR K=1TO6:L(K)=VAL(MID$(DT$(CC(J1
)),K,1))
1815 H4$(J1)=H4$(J1)+HND$(L(K),0):NEXT
K
1820 FOR K=1TO4:RN(K)=VAL(HND$(L(K),0))
:V(K)=RN(K)
1825 IFV(K)>10 THEN V(K)=10
1830 NEXT K
1840 TT=1
1845 FOR K=1TO3:TT=TT+1
1850 IFRIGHT$(HND$(K,0),2)<>RIGHT$(HND$
(K+1,0),2)THEN TT=0:K=3
1855 NEXT K
1860 GOSUB 4630
1900 TT(J1)=TT
1910 OD(J1)=TT
1920 DT=DT-1:R$=DT$(CC(J1)):DT$(CC(J1))
=DT$(DT+1):DT$(DT+1)=R$
1930 NEXT J1
1940 FOR K=1 TO LV:D=1:E=2
1950 FOR L=1 TO LV-1
1960 IF OD(D)<OD(E) THEN F=OD(D):OD(D)=
OD(E):OD(E)=F
1906 D=D+1:E=E+1
1980 NEXT L:NEXT K
1990 FOR J=1TO LV :D=1
2000 IF OD(D)=TT(J) THEN F=J:J=LV
2010 NEXT J:K=0
2020 FOR J=1TO36STEP7:K=K+1
2030 HND$(K,0)=MID$(H4$(F),J,7)
2040 NEXT J
2050 FOR K=5TO6:PL$(K,0)=HND$(K,0):HND$
(K,0)="999":NEXT K
2057 FORJ=1TOLV:H4$(J)="" :NEXT
2060 RETURN
2199 REM *** PUT CARDS IN BOX ***
2200 FOR J=1 TO 2
2210 A=INT((RND(1)*6)+1)*5
2215 IF SCRNC(A,9)=32 THEN 2210

```

```

2220 FORK=9TO11
2230 PLOTA,K,S7$
2240 NEXTK
2250 NEXTJ
2300 FOR PL=5 TO 6
2302 TI$="To put CARDS in BOX press car
d number":CU=25:GOSUB 800
2305 GET C$:C=VAL(C$)
2310 IFC<10RC>6THENTI$="1 to 6 please":
CU=25:GOSUB800:WAIT300:GOTO2302
2320 A=C*5+3
2330 IFSCRNC(A,20)=32THENTI$="Card is in
BOX":CU=25:GOSUB800:WAIT300:GOTO2302
2340 FOR K=17 TO 23
2350 PLOTA-3,K,S7$
2360 NEXT K
2370 PL$(PL,P)=HND$(C,P):HND$(C,P)="999
"
2380 NEXT PL:RETURN
2399 REM **** CUT FOR 5th CARD ****
2400 IFLEAD=FALSETHENTI$="Press SPACE B
AR to cut cards":CU=25:GOSUB800:PF#=0
2410 IFLEAD=TRUETHEN TI$="ORIC to cut":
CU=25:GOSUB800:PF#-P:GOTO2425
2420 GET A$:IF A$<>" " THEN 2400
2425 CT=INT(RND(1)*36)+16
2430 J=0
2435 WAIT 100:FOR J=12 TO 16
2436 PLOT2,J,S7$:NEXT J
2437 FORJJ=13TO15:PLOT5,JJ,18:PLOT2,JJ,
23
2438 PLOT1,JJ,ASC(MID$(CA$(CT),3,1))
2439 NEXTJJ:JJ=0
2450 PLOT 3,13,RIGHT$(CA$(CT),2)
2457 PLOT 3,14,MID$(CA$(CT),4,2)
2458 PLOT 3,15,RIGHT$(CA$(CT),2)
2460 TI$="ORIC PEGS "
2465 IFPF#=PTHENTI$=UU$+" PEGS "
2470 IFVAL(CA$(CT))<>11THENWAIT300:TI$=
"":RETURN
2475 PEG=2:TI$=+TI$+"2 for HIS HEELS":C
U=25:GOSUB800:GOSUB5500
2480 WAIT400:TI$="":RETURN
2599 REM *** PLAY CARDS ***
2600 PL=0:PO=0:F3=FALSE
2605 REPEAT:C$="" :D$="" :F=0:Y3=0:CX=1
:CD=0
2607 GOSUB3100
2608 IF PL=>4 THEN C$="P":LEAD=TRUE
2610 IF LEAD=FALSETHEN2655
2615 IF PO=>4 THEN D$="P":GOTO2655
2620 TI$="ORIC to lead":CU=25:GOSUB800
2621 REM** SET FLAG TO PLAY 5,6,9
2625 FG#=INT(RND(1)*10)+1:F1=1
2630 C=INT(RND(1)*4)+1:IF HND$(C,0)="99
9"THEN 2630

```


CRIBBAGE

```

2635 PO=PO+1:PL$(PO,0)=HND$(C,0):PL$(PO
+6,0)=HND$(C,0)
2637 IF PO=4 THEN2645
2640 IFVAL(PL$(PO,0))<>5ANDVAL(PL$(PO,0
))<>6ANDVAL(PL$(PO,0))<>9THEN2645
2641 IF FG<>4THENPO=PO-1:GOTO2625
2645 GOSUB4300:F=1:REM PLOT ORICCARD
2646 A9=VAL(PL$(PO,0)):IF A9>10 THEN A9
=10
2647 GOSUB3110
2650 GOTO2700
2655 TI$=UU$+" YOUR LEAD":CU=25:GOSUB80
0:WAIT400
2660 A5=1:GOSUB2950
2663 IFC=0THENGOSUB3010:GOTO2663
2665 GOSUB3200 :F1=0
2667 A9=VAL(PL$(PL,P)):IF A9>10 THEN A9
=10
2668 GOSUB3110:A5=0
2700 WAIT200: REPEAT
2710 IF F1=0 THEN 2830
2720 GOSUB 2950 :REM GET C$
2730 IF C$="P" THEN 2830
2740 IF PL=1 AND PO=0 THEN 2800
2750 GOSUB 3400:REM VAL CARDS PLAYED
2800 GOSUB3200: REM PLOT CARD P
2801 GOSUB3120
2810 GOSUB 3600:REM SCORE PLAYED CARD
2815 IF Y3=31 THEN 2881
2820 IF PL=4 AND F=1 THEN2881
2830 IF PO=4 THEN2881
2840 GOSUB 3900 :REM FIND CARD
2845 IF D$="P" THEN TI$="ORIC PASSES":C
U=25:GOSUB800:WAIT200:GOTO2881
2870 GOSUB3400:GOSUB4300:REMPLOT CARD
2875 GOSUB3120
2880 GOSUB 3600
2881 IF PO=4 THEN D$="P"
2883 IF PL=4 THEN C$="P"
2885 IFD$="P"ANDC$="P"ANDY3<31THEN PEG=
1:GOSUB5500
2889 GOSUB3120
2890 F1=1:UNTILC$="P"ANDD$="P"ORY3=31
2895 GOSUB 8000
2897 PLOT28,7," "
2900 UNTIL F3=>8
2910 RETURN
2949 REM *** GET C$ *****
2950 IF C$="P" THEN RETURN
2955 IF PL=4 THEN C$="P":RETURN
2960 LEAD=TRUE
2970 TI$="Press card num. to play "
2975 IFA5<>1THENTI$=TI$+" 'P' to pass"
2980 CU=25:GOSUB 800
2985 C$=""
2990 GET C$:IFC$="" THEN2990

```

```

2995 C=VAL(C$)
3000 IF C>0 AND C<7 THEN 3011
3003 IFC$="P"THENGOSUB3150
3005 IFC$="P"THEN3080
3010 TI$="1 to 6 or 'P' Please":GOTO 29
80
3011 Q=VAL(HND$(C,P)):IFQ>10ANDQ<14THEN
Q=10
3015 A=C*5+1
3020 IFSCRN(A,21)=23THEN3050
3025 IFSCRN(A,18)=23THEN3040
3030 GOSUB3090
3035 TI$="CARD IN BOX Try again":GOTO2
980
3040 GOSUB3090
3045 TI$="CARD PLAYED Try again":GOTO2
980
3050 IFQ+Y3=<31THEN3070
3060 GOSUB3090
3065 TI$="TOTAL OVER 31 Try again":GOT
02980
3070 PL=PL+1:PL$(PL,P)=HND$(C,P):PL$(PL
+6,P)=HND$(C,P)
3080 RETURN
3090 ZAP:WR=INT(RND(1)*7):TI$=WR$(WR):C
U=25:GOSUB 800:WAIT300:RETURN
3100 PLOT 4,7,"TOTAL OF PLAYED CARDS :
":RETURN
3110 A9$=STR$(A9):A9$=MID$(A9$,2):PLOT2
8,7,A9$:RETURN
3120 Y3$=STR$(Y3):Y3$=MID$(Y3$,2):PLOT2
8,7," ":PLOT28,7,Y3$:RETURN
3149 REM CHECK PASS
3150 FOR J=1 TO6:K=VAL(HND$(J,P)):IFK>1
3THEN3170
3155 IFK>10THENK=10
3160 IFK+Y3=>31THEN3170
3165 J=100
3170 NEXT
3175 IFJ<50THEN RETURN
3180 TI$=UU$+" YOU CAN'T PASS":C$="" :CU
=25:GOSUB800:WAIT300:RETURN
3199 REM ** PLOT CARD PLAYER **
3200 IF C=0 THEN RETURN
3205 FOR J=18TO23:PLOTA-1,J,S7$:NEXTJ
3210 FORJJ=17TO19:PLOTA+3,JJ,18:PLOTA,J
J,23
3220 PLOTA-1,JJ,ASC(MID$(HND$(C,P),3,1)
)
3230 NEXTJJ:JJ=0
3260 PLOT A+1,17,RIGHT$(HND$(C,P),2)
3270 PLOT A+1,18,MID$(HND$(C,P),4,2)
3280 PLOT A+1,19,RIGHT$(HND$(C,P),2)
3285 PLOTA,21,0
3290 PLOT A+1,21,C+48:HND$(C,P)="999":F
3=F3+1:PF$=P:CD=CD+1

```

CRIBBAGE

```

3300 RETURN
3399 REM ** VAL PLAYED CARDS **
3400 Z=0:J=0:Z9=0
3410 REPEAT:IFDEAL=TRUE THEN J=J+1
3420 IF DEAL=FALSE AND Z9=0 THEN 3470
3425 IF VAL(PL$(J,P))=0 THEN 3460
3430 Z=Z+1:RN(Z)=0:V(Z)=0
3440 RN(Z)=VAL(PL$(J,P))
3460 IF Z=CD+1 THEN 3510
3470 IF DEAL=FALSE THEN J=J+1:Z9=1
3475 IF VAL(PL$(J,O))=0 THEN 3510
3480 Z=Z+1:RN(Z)=0:V(Z)=0
3490 RN(Z)=VAL(PL$(J,O))
3510 UNTIL Z=CD+1
3511 FOR J=1 TO Z:IF RN(J)<>0 THEN 3513
3512 R1=RN(J):RN(J)=RN(J+1):RN(J+1)=R1
3513 V(J)=RN(J):IF V(J)>10 THEN V(J)=10
3514 NEXT J
3520 J=0:Y3=0
3530 REPEAT
3540 J=J+1:Y1=V(J)
3550 Y3=Y3+Y1
3560 UNTIL J=Z OR Y3=>31
3570 RETURN
3599 REM ** SCORE PLAYED CARDS **
3600 Z=CD
3610 IF Y3=15 THEN PEG=2:GOSUB 5500
3620 IF Y3=31 THEN PEG=2:GOSUB 5500
3630 PEG=0
3699 REM *PAIR AND RUN*
3700 IF RN(Z)<>RN(Z-1) THEN 3745
3710 IF RN(Z-1)<>RN(Z-2) THEN PEG=2:GOTO 3740
3720 IF RN(Z-2)<>RN(Z-3) THEN PEG=6:GOTO 3740
3730 PEG=12
3740 GOSUB 5500:PEG=0:RETURN
3745 IF CD<3 THEN RETURN
3750 FOR J=CX TO CD:RC(J)=RN(J):NEXT J
3760 FOR J=CX TO CD:FOR K=CX TO CD-1
3770 IF RC(K)>RC(K+1) THEN R3=RC(K):RC(K)=
RC(K+1):RC(K+1)=R3
3780 NEXT K:NEXT J
3785 PEG=0
3790 FOR J=CX TO CD-1
3795 IF RC(J)=0 THEN 3810
3800 IF RC(J)=RC(J+1)-1 THEN PEG=PEG+1
3805 XX=XX+1
3810 NEXT J
3820 IF PEG=XX THEN PEG=PEG+1:GOSUB 5500:GO
TO 3840
3830 CX=CX+1
3840 FOR J=1 TO 8:RC(J)=0:RN(J)=0:NEXT J:P
EG=0:XX=0
3850 RETURN
3899 REM ** FIND CARD ORIC **

```

```

3900 REM: FOR J=1 TO 5:PING:WAIT 20:NEXT
3905 IF D$="P" THEN RETURN
3910 LEAD=FALSE
3915 TI$="ORIC to play":CU=25:GOSUB 800:
WAIT 200
3920 C=0
3930 REPEAT:C=C+1:I=VAL(HND$(C,O))
3935 IF I>10 AND I<14 THEN I=10
3940 UNTIL Y3+I=31 OR C=4
3950 IF Y3+I=31 THEN 4240
3960 C=0
3970 REPEAT:C=C+1:I=VAL(HND$(C,O))
3975 IF I>10 AND I<14 THEN I=10
3980 UNTIL Y3+I=15 OR C=4
3990 IF Y3+I=15 THEN 4240
4010 C=0
4020 REPEAT:C=C+1:I=VAL(HND$(C,O))
4025 IF I>10 AND I<14 THEN I=10
4030 UNTIL VAL(PL$(PL,P))=VAL(HND$(C,O)
) OR C=4
4040 IF VAL(PL$(PL,P))=VAL(HND$(C,O)) AN
DY3+I=<31 THEN 4240
4050 C=0
4060 REPEAT:C=C+1:I=VAL(HND$(C,O))
4065 IF I>10 AND I<14 THEN I=10
4070 UNTIL VAL(PL$(PL,P))=VAL(HND$(C,O)
)+1 OR C=4
4080 IF VAL(PL$(PL,P))=VAL(HND$(C,O))+1
AND Y3+I=<31 THEN 4240
4090 C=0
4100 REPEAT:C=C+1:I=VAL(HND$(C,O))
4105 IF I>10 AND I<14 THEN I=10
4110 UNTIL VAL(PL$(PL,P))=VAL(HND$(C,O)
)-1 OR C=4
4120 IF VAL(PL$(PL,P))=VAL(HND$(C,O))-1
AND Y3+I=<31 THEN 4240
4140 C=0
4150 REPEAT:C=C+1:I=VAL(HND$(C,O))
4155 IF I>10 AND I<14 THEN I=10
4160 UNTIL VAL(HND$(C,O))>10 AND I+Y3=<
31 OR C=4
4170 IF I+Y3=<31 THEN 4240
4180 C=0
4183 REPEAT:C=C+1:I=VAL(HND$(C,O))
4185 IF I>10 AND I<14 THEN I=10
4190 UNTIL I+Y3=<31 OR C=4
4200 IF I+Y3=<31 THEN 4240
4230 D$="P":GOTO 4260
4240 PO=PO+1:PL$(PO,O)=HND$(C,O):PL$(PO
+6,O)=HND$(C,O)
4260 RETURN
4299 REM PLOT ORIC CARD
4300 A=INT((RND(1)*6)+1)*5+1
4310 IF SCRN(A,10)<>64 THEN 4300
4320 FOR J=8 TO 12:PLOTA-1,J,S7$:NEXT J
4330 FOR JJ=9 TO 11:PLOTA+3,JJ,18:PLOTA,JJ

```

CRIBBAGE

```
,23
4335 PLOTA-1,JJ,ASC(MID$(HND$(C,0),3,1)
)
4340 NEXTJJ:JJ=0
4360 PLOTA+1,9,RIGHT$(HND$(C,0),2)
4365 PLOTA+1,10,MID$(HND$(C,0),4,2)
4366 PLOTA+1,11,RIGHT$(HND$(C,0),2)
4370 HND$(C,0)="999":F3=F3+1:PF*=0:CD=C
D+1
4380 RETURN
4499 REM *** SCORE PLAYED HAND ***
4500 PEG=1:FOR J=1TO3
4502 IFRIGHT$(PL$(J,PF*),2)=RIGHT$(PL$(
J+1,PF*),2)THENPEG=PEG+1
4504 NEXT J
4506 IF PEG<>4THEN 4550
4508 IFRIGHT$(PL$(1,PF*),2)=RIGHT$(CA$(
CT),2) THEN PEG=PEG+1
4510 GOSUB5200:TI$=TI$+" FOR FLUSH"
4512 CU=25:GOSUB800:WAIT200
4514 GOSUB5500:WAIT200
4550 PEG=0
4552 FOR J=1 TO 4
4553 RN(J)=VAL(PL$(J,PF*))
4554 U(J)=RN(J):IF U(J)>10THENU(J)=10
4555 NEXT J:GOTO 4620
4559 REM *VAL BOX HAND*
4560 PEG=1
4562 FOR J=5TO6:FOR K=0TO1
4564 IFRIGHT$(CA$(CT),2)=RIGHT$(PL$(J,K
),2)THENPEG=PEG+1
4566 NEXT K:NEXT J
4568 IF PEG<>5 THEN 4600
4570 GOSUB5200:TI$=TI$+" FOR FLUSH"
4572 CU=25:GOSUB800:WAIT200
4574 GOSUB5500:WAIT200
4600 PEG=0
4610 K=4 :FOR J=1 TO 4:K=K+1
4611 RN(J)=VAL(PL$(K,0))
4612 U(J)=RN(J):IF U(J)>10THENU(J)=10
4613 J=J+1:RN(J)=VAL(PL$(K,P))
4614 U(J)=RN(J):IF U(J)>10THENU(J)=10
4615 NEXT J
4619 REM *VAL CUT CARD*
4620 J=5:RN(J)=VAL(CA$(CT)):U(J)=RN(J):
IF U(J)>10THENU(J)=10
4625 GOTO 4640
4628 REM*RN(J),U(J) TO 100 SORT ORIC
4629 REM HAND ONLY*
4630 RN(5)=100:U(5)=100
4639 REM SCORE 15's
4640 PEG=0:F=0
4650 FOR J=1TO5:F=F+U(J):NEXT J
4660 IF F=15 THEN PEG=2:GOTO 4810
4670 FOR M=1 TO 2
4680 :FOR L=M+1 TO 3
```

```
4690 : FOR K=L+1 TO 4
4700 : FOR J=K+1 TO 5
4710 IF U(M)+U(L)+U(K)+U(J)=15 THEN PEG
=PEG+2
4720 NEXT J:NEXT K:NEXT L:NEXT M
4750 FOR L=1 TO 3:FOR K=L+1 TO4:FOR J=K
+1 TO 5
4760 IF U(L)+U(K)+U(J)=15 THEN PEG=PEG+
2
4770 NEXT J:NEXT K:NEXT L
4780 FOR K=1 TO 4:FOR J=K+1 TO 5
4790 IF U(K)+U(J)=15 THEN PEG=PEG+2
4800 NEXTJ:NEXT K
4810 TT=TT+PEG
4820 IF RN(5)=100 THEN 4860
4825 IF PEG=0 THEN 4860
4830 GOSUB 5200
4840 TI$=TI$+" FOR 15's"
4850 CU=25:GOSUB 800:WAIT 200
4855 GOSUB 5500:WAIT200
4859 REM SCORE PAIR
4860 PEG=0
4870 FOR K=1 TO 4:FOR J=K+1 TO 5
4880 IF RN(K)=RN(J) THEN PEG=PEG+2
4890 NEXT J:NEXT K
4900 TT=TT+PEG
4910 IF RN(5)=100 THEN 4940
4915 IF PEG=0 THEN 4940
4920 GOSUB 5200:TI$=TI$+" FOR PAIRS"
4930 CU=25:GOSUB 800:WAIT 200
4935 GOSUB 5500:WAIT200
4938 REM *** SCORE RUN ***
4939 REM SORTE INTO NUM ORDER FIRST
4940 FOR K=1 TO 5:FOR J=1 TO 4
4950 IF RN(J)>RN(J+1) THEN F=RN(J):RN(J
)=RN(J+1):RN(J+1)=F
4960 NEXT J:NEXT K:PEG=0
4970 IFRN(1)<>RN(2)-1ORRN(2)<>RN(3)-1OR
RN(3)<>RN(4)-1ORRN(4)<>RN(5)-1THEN5000
*4980 PEG=5:GOTO5100
5000 FOR M=1 TO 2:FOR L=M+1 TO 3:FOR K=
L+1 TO 4:FOR J=K+1 TO 5
5040 IFRN(M)=RN(L)-1ANDRN(L)=RN(K)-1AND
RN(K)=RN(J)-1THENPEG=PEG+4
5050 NEXT J:NEXT K:NEXT L:NEXT M
5060 IF PEG<>0 THEN 5100
5070 FOR L=1 TO3:FOR K=L+1 TO4:FOR J=K+
1 TO5
5080 IFRN(L)=RN(K)-1ANDRN(K)=RN(J)-1THE
NPEG=PEG+3
5090 NEXT J:NEXT K:NEXT L
5100 TT=TT+PEG
5110 IF RN(5)=100 THEN5160
5112 IF PEG=0 THEN 5120
5114 GOSUB 5200:TI$=TI$+" FOR RUN'S"
5116 CU=25:GOSUB 800:WAIT 200
```

CRIBBAGE

```

5118 GOSUB 5500:WAIT200:PEG=0
5120 IF BOX#=0 THEN 5142
5122 FOR J=5TO6:FOR K=0TO1
5124 IFRIGHT$(CA$(CT),2)=RIGHT$(PL$(J,K
),2)ANDVAL(PL$(J,K))=11THENPEG=1
5126 NEXT K:NEXT J
5130 BOX#=0:GOTO5150
5142 PEG=0
5144 FOR K=1TO4
5146 IFRIGHT$(PL$(K,PF#),2)=RIGHT$(CA$(
CT),2)ANDVAL(PL$(K,PF#))=11THENPEG=1
5148 NEXT K
5150 IF PEG=0 THEN 5158
5152 TI$="ONE for his NIB'S":CU=25
5154 GOSUB800:WAIT200
5156 GOSUB5500:WAIT200
5158 RETURN
5160 FOR J=1TO5:RN(J)=0:V(J)=0:NEXT J
5170 RETURN
5200 TI$=UU$+" YOUR "
5210 IF PF#=0 THEN TI$="ORIC'S "
5220 PEG$=STR$(PEG):PEG$=MID$(PEG$,2)
5230 TI$=TI$+"SCORE: "+PEG$
5240 RETURN
5299 REM *** SHOW BOX+GOSUB SCORE ***
5300 TI$="":CU=25:GOSUB800
5304 PLOT4,7,S4$
5305 FORJ=8TO26
5310 PLOT 6,J,S3$
5320 NEXT J
5325 C=4:F=0:BOX#=1
5330 IFDEAL=TRUETHENPLOT14,7,"ORIC'S BO
X":GOTO5365
5335 PLOT12,7,UU$+"S BOX"
5365 FOR CE=9 TO 30 STEP 7:C=C+1
5370 FORJJ=13TO15:PLOTCE+4,JJ,18:PLOTCE
+1,JJ,23
5375 PLOTCE,JJ,ASC(MID$(PL$(C,F),3,1))
5380 NEXTJJ:JJ=0
5395 PLOT CE+2,13,RIGHT$(PL$(C,F),2)
5400 PLOT CE+2,14,MID$(PL$(C,F),4,2)
5410 PLOT CE+2,15,RIGHT$(PL$(C,F),2)
5420 IF C=6 THEN C=4:F=P
5430 NEXT CE
5440 GOSUB 4560:WAIT400
5445 PLOT4,7,S4$:FORJ=8TO26:PLOT2,J,SP$
:NEXTJ
5450 RETURN
5499 REM *** PEG ***
5500 IF PF#<>P THEN 5600
5510 SC(P)=SC(P)+PEG
5515 IF PEG=0 THEN RETURN
5520 Q1=XP:Q2=OP:Q3=SC(P)
5530 Q4=L1:Q5=L2:Q6=L3:Q7=1:Q8=3
5535 IFBELLTHENPING
5540 GOSUB 5700

```

```

5550 XP=Q2:OP=Q3:L1=Q4:L2=Q5:L3=Q6
5560 IF SC(P)=>121 THEN 410
5570 PEG=0:RETURN
5600 SC(O)=SC(O)+PEG
5605 IF PEG=0 THEN RETURN
5610 Q1=XO:Q2=OO:Q3=SC(O)
5620 Q4=L4:Q5=L5:Q6=L6:Q7=6:Q8=4
5625 IFBELLTHENPING
5630 GOSUB 5700
5640 XO=Q2:OO=Q3:L4=Q4:L5=Q5:L6=Q6
5650 IF SC(O)=>121 THEN 410
5660 PEG=0:RETURN
5700 IF Q3<91 THEN 5760 ELSE Q3=Q3-89
5710 IF Q5=Q8THEN Q4=Q8
5720 IF Q6=Q8THEN Q5=Q8
5730 Q6=Q8
5740 GOSUB 5860
5750 RETURN
5760 IF Q3<61 THEN 5810 ELSE Q3=Q3-60
5770 IF Q5=Q7THEN Q4=Q7
5780 IF Q6=Q7THEN Q5=Q7
5790 Q6=Q7:Q3=37-Q3
5800 GOSUB 5860: RETURN
5810 IF Q3<31 THEN 5840 ELSE Q3=Q3-29
5820 GOSUB 5710
5830 RETURN
5840 GOSUB 5790
5850 RETURN
5860 IF Q6=Q8THEN GOSUB 5920 ELSE GOSUB
5950
5870 IF SC(P)=>121 OR SC(O)=>121 THEN Q
3=36
5880 PLOT Q1,Q4," "
5890 PLOT Q2,Q5,"J"
5900 PLOT Q3,Q6,"J"
5910 RETURN
5919 REM * JUMP SPACE FROM L-R
5920 FOR J=6 TO 30 STEP 6
5930 IF Q3>J THEN Q3=Q3+1
5940 NEXT J:RETURN
5949 REM * JUMP SPACE FROM R-L
5950 FOR J=32 TO8 STEP-6
5960 IF Q3<J THEN Q3=Q3-1
5970 NEXT J:RETURN
5999 REM *** SHOW WINER + NEW GAME ***
6000 WAIT100:CLS:FORJ=0TO7:PAPERJ:WAIT2
0:NEXTJ
6020 TI$="ORIC IS THE WINNER":K=1:PAPER
6:INK0
6030 IFSC(P)>=121THENTI$="YOU ARE THE W
INNER":PLOT14,14,UU$:K=2:PAPER1:INK7
6040 CU=12:GOSUB 800
6050 FORJ=1TO23
6060 NU=INT(ASC(MID$(MU$(K),J,1))-64)
6070 WT=INT(ASC(MID$(WT$(K),J,1))-64)
6080 OC=3:IFNU>12THENOC=4:NU=NU-12

```


CRIBBAGE

```

6090 MUSIC 2,0C,NU,10
6100 MUSIC 1,0C+3,NU,0
6110 MUSIC 3,0C-1,NU,8
6120 WAITWT*12
6130 PLAY 7,0,1,3000
6140 NEXTJ
6150 PLAY0,0,0,0
6240 CLS:WAIT200:PLOT14,14,UU$:WAIT100
6250 TI$="PLAY AGAIN"
6260 CU=16:GOSUB 800:WAIT100
6270 GOSUB 480
6280 IF A$<>"Y" THEN RETURN
6290 RUN20
6500 FORX=14TO24
6510 UU=SCRN(X,14)
6520 UU$=UU$+CHR$(UU)
6530 IFUU=32ANDSCRN(X+1,14)=32THENX=35
6540 NEXTX
6550 X=0:UU=0
6570 UU$=LEFT$(UU$,LEN(UU$)-1)
6580 RETURN
6998 REM***** IN.UAL.*****
6999 REM***** DIM ARRAYS*****
7000 DIM CC(15),RN(10),U(10),L(6)
7010 DIM TT(15),OD(15),CT(1),SC(1)
7015 DIM RC(8)
7020 DIM CA$(52),HND$(12,1),PL$(12,1)
7030 DIM DT$(15),H4$(15),S$(13)
7039 REM *** MAKE CARDS ***
7040 FOR J=1 TO 8:READ R
7060 FOR K=0 TO 7:READ S
7080 POKE #B400+8*R+K,S
7090 NEXT K:NEXT J
7100 DATA 92,1,7,15,15,13,1,3,7
7101 DATA 94,0,48,56,56,24,0,32,48
7102 DATA 95,3,7,3,29,63,29,1,3
7103 DATA 106,32,48,32,28,62,28,0,32
7104 DATA 122,12,30,31,31,15,7,3,1
7105 DATA 123,24,60,60,60,56,48,32,0
7106 DATA 124,1,3,7,15,15,7,3,1
7107 DATA 125,0,32,48,56,56,48,32,0
7120 FOR J=1 TO 13:READ S$(J):NEXT J
7130 DATA"1 A ","2 2 ","3 3 ","4 4 ","5
5 ","6 6 ","7 7 ","8 8 ","9 9 ","1010"
7131 DATA"11J ","12Q ","13K "
7160 FOR J=1 TO 13
7170 CA$(J)=LEFT$(S$(J),2)+CHR$(0)+RIGH
T$(S$(J),2)+"^"
7180 CA$(J+13)=LEFT$(S$(J),2)+CHR$(0)+R
IGHT$(S$(J),2)+"_ "
7190 CA$(J+26)=LEFT$(S$(J),2)+CHR$(1)+R
IGHT$(S$(J),2)+"z("
7200 CA$(J+39)=LEFT$(S$(J),2)+CHR$(1)+R
IGHT$(S$(J),2)+"!)"
7205 NEXT J
7209 REM ** MAKE PEG HOLES +PEGS **

```

```

7210 FOR J=1 TO 2:READ R
7220 FOR K=0 TO 7:READ S
7230 POKE #B400+8*R+K,S
7240 NEXT K:NEXT J
7250 DATA 33,12,12,0,0,0,0,12,12
7251 DATA 93,4,14,21,4,4,21,14,4
7300 FORJ=0TO7
7310 READA:POKE46592+J,63-A
7320 NEXTJ
7330 DATA33,18,12,33,33,12,18,33
7350 A=0:B=0
7378 REM ** $+Num UAL **
7379 REM** 37 SPACES **
7380 SP$="
"
7384 REM** 27 SPACES **
7385 S4$="
"
7389 REM** 34 SPACES **
7390 S3$="
"
7399 REM** 3 SPACES AND 5 SPACES **
7400 S5$=" " :S7$=" "
7410 P=1:O=0
7415 L1=1:L2=1:L3=1:L4=6:L5=6:L6=6
7420 XP=37:OP=37:X0=37:O0=37
7430 FORJ=1TO15:READDT$(J):NEXTJ
7440 DATA 123456,123546,123645,124536,1
24635,125634,134526,134625
7441 DATA 135624,145623,234516,234615,2
35614,245613,345612
7500 MU$(1)="QOMLMHJLMJHEQOMLMHJFEHM"
7510 WT$(1)="AAAABCAAAABCAAAAAAABCD"
7520 MU$(2)="CEFCEHJLMJHEMLMHJHJFEHM"
7530 WT$(2)="AAAABCAAAABCAAAAAAABDD"
7600 FORJ=0TO7
7610 PC$(J)=CHR$(144+J)
7620 IC$(J)=CHR$(128+J)
7630 NEXTJ
7640 PC$(8)=CHR$(138)
7650 IC$(8)=CHR$(142)
7700 FORJ=0TO6:READWR$(J):WR$(J)=CHR$(1
2)+WR$(J):NEXTJ
7710 DATAWRONG...,CHEAT...,DUM - DUM,HE
Y..That's not right
7711 DATAEre..you can't do that,ERE..I
think that's wrong
7712 DATAAre you trying to confuse me
7900 RETURN
8000 FOR J=1 TO 4
8010 PL$(J,P)=" "
8020 PL$(J,O)=" "
8030 NEXT J
8040 RETURN
8100 FOR J=1TO4
8110 PL$(J,P)=PL$(J+6,P)
8120 PL$(J,O)=PL$(J+6,O)

```

CRIBBAGE

```

8130 NEXT J
8140 RETURN
9000 CLS:TI$="*****":CU=3:GOSUB
800
9005 TI$="LEVEL OF PLAY":CU=2:GOSUB800:
WAIT300
9010 TI$="There are 5 levels of play":C
U=CU+4:GOSUB800:WAIT200
9020 TI$="Please enter level required":
CU=CU+5:GOSUB800
9030 TI$="by pressing a key from 1 to 5
":CU=CU+2:GOSUB800
9040 K$=KEY$:X=RND(1):IFK$=""THEN9040
9045 IFVAL(K$)>0ANDVAL(K$)<6THEN9060
9050 TI$="ONLY 1 TO 5":CU=CU+2:GOSUB800
:GOTO9040
9060 LV=VAL(K$):LV=LV*3
9070 WAIT100:TI$="LEVEL CHOSEN IS "+K$:
CU=CU+5:GOSUB800:WAIT300
9080 RETURN
10000 CLS:PAPER 4:INK 7
10030 FORK=1TO17:PRINT:NEXTK
10040 PLOT12,9,B5$:PLOT12,14,B5$
10045 A$=CHR$(4)+CHR$(27)
10050 PRINTSPC(6)A$"NC R I B M A S T E
R"
10060 PRINT CHR$(4)
10100 K=3:L=9
10110 REPEAT
10120 FORJ=4TO10
10130 PLOTL,J,20
10140 PLOTK,J,23
10150 NEXT J
10160 K=K+9:L=L+9
10170 UNTILK>30
10175 PLOT2,5,0:PLOT2,9,0
10180 FORK=4TO32STEP9
10190 PLOTK,5,"A"
10200 PLOTK+ 3,9 ,"A"
10210 NEXTK
10220 PLOT4,7,CHR$(1)+"z{"
10230 PLOT13,7,CHR$(0)+" _j"
10240 PLOT22,7,CHR$(1)+"{|}"
10250 PLOT31,7,CHR$(0)+"\^"
10890 FORJ=1TO23
10900 NU=INT(ASC(MID$(MU$(1),J,1))-64)
10910 WT=INT(ASC(MID$(WT$(1),J,1))-64)
10920 OC=3:IFNU>12THENOC=4:NU=NU-12
10930 MUSIC 2,OC,NU,10
10940 MUSIC 1,OC+3,NU,0
10950 MUSIC 3,OC-1,NU,8
10960 WAITWT*12
10970 PLAY 7,0,1,3000
10980 IFJ<18THEN11010
10990 L=43-J
10995 A$="by THE EMMIT"

```

```

11000 PLOT12,L,3:PLOT13,L,A$:PLOT12,L+1
," "
11010 NEXTJ
11020 PLAY0,0,0,0
11100 WAIT400:CLS:INK3
11105 POKE48000,23:POKE48001,0
11110 B$="CRIB MASTER by THE
EMMIT"
11120 FOR K=1 TO LEN(B$):A=ASC(MID$(B$,
K,1)):POKE48001+K,A:NEXT K
11125 POKE48036,9:POKE48037,66:POKE4803
8,66:POKE48039,8
11210 TI$="Please enter your name":CU=8
:GOSUB800
11215 TI$="then press <RETURN>":CU=11:G
OSUB800
11220 FORJ=1TO14:PRINT:NEXTJ
11225 INPUTUU$
11230 IFLEN(UU$)<11THEN11240
11232 TI$="NAME TO LONG":CU=CU+9:GOSUB8
00:WAIT500:CLS:GOTO11210
11240 WAIT100:CLS
11250 TI$="I'm ORIC pleased to meet you
":CU=8:GOSUB800
11260 TI$=UU$:CU=11:GOSUB800
11270 WAIT200:TI$="So you want to play
me at CRIB":CU=14:GOSUB800
11280 WAIT800:CLS
11990 RETURN
12000 CLS:PAPER0:INK2
12002 FORJ=1TO7:PRINT:NEXTJ
12004 PRINTPC$(8)IC$(4)PC$(6)"*****
*****"PC$(0)
12005 PRINTPC$(8)IC$(4)PC$(6)"*****
*****"PC$(0)
12006 PRINT:PRINT" THIS IS SIX CARD CR
IBBAGE FOR TWO":PRINT
12007 PRINTSPC(15)"PLAYERS":PRINT
12008 PRINT:PRINTSPC(8)"ORIC IS YOUR OP
PONENT"
12010 PRINT
12014 PRINTPC$(8)IC$(4)PC$(6)"*****
*****"PC$(0)
12015 PRINTPC$(8)IC$(4)PC$(6)"*****
*****"PC$(0)
12016 GOSUB20000:CLS
12018 PRINT:PRINT:PRINTSPC(10)PC$(7)IC$
(1)"THE CRIB BOARD "PC$(0)
12022 PRINT:PRINT" The holes at the ext
reme right are"
12024 PRINT"where the"IC$(1)"PEGS"IC$(2
)"mark 0 (Zero) at the"
12026 PRINT"start, and they are subsequ
ently the final holes in the game."
12028 PRINT:PRINT" The score is recorde
d by the pegs"

```

CRIBBAGE

```

12030 PRINT"which are moved from right
to left,"
12032 PRINT"then back along the inner '
track',"
12034 PRINT"then around the 'track' onc
e more"
12036 PRINT"until the score is"IC$(1)"1
21"IC$(2)"or"IC$(1)"OVER"
12038 PRINT:PRINT" Each player has his
own 'track'"
12040 PRINT"and"IC$(1)"2"IC$(2)"pegs. E
ach new score is marked"
12042 PRINT"on the board by taking out
the"
12044 PRINT"trailing peg and placing it
in front"
12046 PRINT"of the leading peg by the a
ppropriate"
12048 PRINT"number of points (score)"
12052 GOSUB20000:CLS
12054 PRINT:PRINTSPC(10)PC$(3)IC$(0)"IN
STRUCTIONS "PC$(0)
12058 PRINT:PRINT" This game is played
to a score of"
12060 PRINTIC$(5)"121 'PEGS' (points)"I
C$(2)"or more to win"
12062 PRINT:PRINT
12064 PRINT" Both players cut the cards
for the"
12066 PRINT"deal - highest card deals."
:PRINT
12068 PRINTPC$(1)IC$(0)SPC(6)" ACE has
a value of 1"
12070 PRINTPC$(1)IC$(0)" COURT cards
( J,Q,K ) value 10"
12072 PRINT:PRINTPC$(3)IC$(0)"THE DEAL
:- "PC$(0):PRINT
12074 PRINTIC$(5)"6"IC$(2)"cards are de
alt to each player face down"
12076 PRINT:PRINTPC$(3)IC$(0)"THE BOX
:- "PC$(0):PRINT
12078 PRINT" Look at your hand and sele
ct"IC$(5)"2"IC$(2)"cards"
12079 PRINT"to put in the"IC$(5)"BOX"IC
$(2)"face down."IC$(5)"ORIC"
12080 PRINT"does likewise. At the end o
f the play"
12082 PRINT" the"IC$(5)"BOX"IC$(2)"belo
ngs to the"IC$(5)"DEALER"
12084 GOSUB20000:CLS
12086 PRINT:PRINTPC$(3)IC$(0)"THE START
ER :- "PC$(0):PRINT
12088 PRINT" The remaining cards are cu
t by the"
12090 PRINT"non-dealer, and the dealer
takes the"

```

```

12092 PRINT"top card from the cut and t
urns it"
12094 PRINT"over as the"IC$(5)"STARTER"
IC$(2)"card. If the"
12096 PRINT"starter card is a"IC$(5)"JA
CK"IC$(2)"the dealer"
12098 PRINTIC$(5)"PEGS 2"IC$(2)"points"
IC$(1)"' for his HEELS'"
12100 PRINT:PRINTPC$(3)IC$(0)"THE PLAY
:- "PC$(0):PRINT
12102 PRINT"The non-dealer plays the fi
rst card"
12104 PRINT"followed by the dealer. The
face"
12106 PRINT"values of the cards are add
ed"
12108 PRINT"together and called as they
are"
12110 PRINT"played, but the culminating
total"
12112 PRINT"must not exceed"IC$(5)"31."
IC$(2)"If you find"
12114 PRINT"you cannot play because the
total"
12116 PRINT"will exceed"IC$(5)"31"IC$(2
)"you"IC$(0)PC$(3)"MUST PASS."PC$(0)
12118 PRINT"If the opponent can still p
lay a card"
12120 PRINT"he must play until all his
cards are"
12122 PRINT"laid, or he too is obliged
to"IC$(0)PC$(3)"PASS."PC$(0)
12124 GOSUB20000:CLS
12126 PRINT:PRINT:PRINT:PRINT:PRINT
12128 PRINT:PRINT" If either player st
ill has cards"
12130 PRINT"left in his hand, another s
eries is"
12132 PRINT"played up to"IC$(5)"31,"IC$
(2)"starting with the"
12134 PRINT"player whose turn falls nex
t."
12136 PRINT:PRINT:PRINT" During the pla
y certin scores can"
12138 PRINT"be"IC$(5)"PEGGED"
12140 GOSUB20000:CLS
12142 PRINT:PRINTSPC(5)PC$(3)IC$(1)"SCO
RING DURING THE PLAY "PC$(0):PRINT
12144 PRINTIC$(1)" A)"IC$(2)" If the
exact total of"IC$(3)"15"IC$(2)"can"
12146 PRINT"be reached, then"PC$(7)IC$(
1)"PEG '2 for 15'"PC$(0):PRINT:PRINT
12148 PRINTIC$(1)" B)"IC$(2)" If the
card played is the same";
12150 PRINT"rank as the last card playe
d then"

```

CRIBBAGE

```

12152 PRINTPC$(7)IC$(1)"PEG '2 for a PA
IR'"PC$(0):PRINT:PRINT
12154 PRINTIC$(1)" C)"IC$(2)" If the
card played is the same";
12156 PRINT"rank as the last"IC$(3)"2"IC
C$(2)"cards played then"
12158 PRINTPC$(7)IC$(1)"PEG '6 for a PA
IR ROYAL'"PC$(0):PRINT:PRINT
12160 PRINTIC$(1)" D)"IC$(2)" If the
card played is the same";
12162 PRINT"rank as the last"IC$(3)"3"IC
C$(2)"cards played then"
12164 PRINTPC$(7)IC$(1)"PEG '12 for a D
OUBLE PAIR ROYAL'"PC$(0)
12166 GOSUB20000:CLS
12168 PRINT:PRINTIC$(1)" E)"IC$(2)" I
f the card played taken in"
12170 PRINT"relation to the immediately
"
12172 PRINT"preceeding cards, forms a"IC
C$(3)"RUN of 3"
12174 PRINT"or more cards, then"
12176 PRINTPC$(7)IC$(1)"PEG '1' for eac
h card in the RUN"PC$(0)
12178 PRINT:PRINTIC$(1)" F)"IC$(2)" I
f the card played reaches"IC$(3)"31"
12180 PRINT"exactly then"
12182 PRINTPC$(7)IC$(1)"PEG '2' for a G
O"PC$(0)
12184 PRINT:PRINTIC$(1)" G)"IC$(2)" T
he last player who brings"
12186 PRINT"the total closest to"IC$(3)
"31"
12188 PRINTPC$(7)IC$(1)"PEGS '1' for a
GO"PC$(0)
12190 GOSUB20000:CLS
12192 PRINT:PRINT:PRINT:PRINT:PRINT:PRI
NT:PRINT:PRINT" When there are no more
cards to"
12194 PRINT"play, the players then scor
e their"
12196 PRINT"individual hands, the deale
r last."
12198 PRINT"At this stage the"IC$(5)"ST
ARTER"IC$(2)"card is"
12200 PRINT"used to form a"IC$(5)"5 car
d hand"IC$(2)"for each"
12202 PRINT"player and a"IC$(5)"5 card
hand"IC$(2)"for the BOX,"
12204 PRINT"which belongs to the"IC$(5)
"DEALER"
12206 GOSUB20000:CLS
12208 PRINT:PRINTSPC(8)PC$(1)IC$(6)"SCO
RING THE HANDS "PC$(0)
12210 INK4
12212 PRINT:PRINT" 15's

```

```

:"IC$(0)PC$(3)"PEG 2"
12214 PRINT:PRINT" PAIRS
:"IC$(0)PC$(3)"PEG 2"
12216 PRINT:PRINT" PAIR ROYAL
:"IC$(0)PC$(3)"PEG 6"
12218 PRINT:PRINT" DOUBLE PAIR ROYAL
:"IC$(0)PC$(3)"PEG 12"
12220 PRINT:PRINT" RUNS (3 or more ca
rds) : "IC$(0)PC$(3)"PEG 1"
12222 PRINTSPC(27)IC$(0)PC$(3)"per card
"
12224 PRINT:PRINT" FLUSH in HAND
:"IC$(0)PC$(3)"PEG 4"
12226 PRINT:PRINT" FLUSH in HAND + STA
RTER : "IC$(0)PC$(3)"PEG 5"
12228 PRINT:PRINT" FLUSH in BOX + STAR
TER : "IC$(0)PC$(3)"PEG 5"
12230 PRINT:PRINT" JACK (of the same
suit"
12232 PRINT" as the STARTER) :
"IC$(0)PC$(3)"PEG 1 for";
12234 PRINTSPC(27)IC$(0)PC$(3)"his NIB'
S"
12236 GOSUB20000:CLS
12238 PRINT:PRINT:PRINTSPC(11)PC$(6)IC$
(0)"CONTROL KEYS "PC$(0)
12242 PRINT:PRINT:PRINT:PRINT
12244 PRINTIC$(1)" ANY KEY : CUT
S CARDS FOR DEAL":PRINT
12246 PRINTIC$(2)" 'P' : PAS
S":PRINT:PRINT
12248 PRINTIC$(3)" 1 TO 6 : PLA
YS or PUTS CARD"
12250 PRINTIC$(3)SPC(18)"IN BOX":PRINT
12252 PRINT" SPACE BAR : CUTS CAR
DS IN PLAY":PRINT:PRINT
12254 PRINTPC$(6)IC$(4)" DO YOU WANT A
PING WHEN PEGGING "PC$(0)
12256 CU=20:GOSUB480:BELL=FALSE
12258 IFA$="Y"THEN BELL=TRUE
12260 CLS:FORJ=1TO9:PRINT:NEXTJ
12262 PRINTIC$(8)IC$(5)SPC(6)"DO YOU WA
NT TO READ THE"
12263 PRINTIC$(8)IC$(5)SPC(6)"DO YOU WA
NT TO READ THE"
12264 PRINTIC$(8)IC$(5)SPC(9)"INSTRUCTI
ONS AGAIN"
12265 PRINTIC$(8)IC$(5)SPC(9)"INSTRUCTI
ONS AGAIN"
12266 CU=19:GOSUB480
12268 IFA$="Y"THEN12000
12270 RETURN
20000 PLOT1,25,12:PLOT2,25,1:PLOT6,25,"
Press ANY KEY to continue"
20005 REPEAT:X=RND(9):UNTILKEY$<>"
20010 RETURN

```


PITFALL

```

1 PING
2 CLS
3 PAPER0:INK2
4 PLOT14,9,CHR$(10)+"PITFALL
5 PLOT14,10,CHR$(10)+"PITFALL
10 DATAA2,17,A0,00,BD,15,BD,A8,C0,64,F0,
8,CA,E0,00
11 DATAD0,F3,4C,22,98
12 DATAA9,20,9D,15,BD,98,CA,9D,15,BD,CA,
4C,0D,98
13 DATAAE,23,BD,E0,20,F0,0,4C,31,98
15 DATAA9,61,8D,23,BD
16 DATAAE,15,BD,8E,B5,BD,A2,20,8E,15,BD
17 DATAA2,00,A0,00,BD,B5,BD,A8,C0,64,F0,
8
18 DATAE8,E0,1F,D0,F3,4C,5E,98
20 DATAA9,20,9D,B5,BD,98,E8,9D,B5,BD,E8,
4C,49,98
21 DATAAE,B7,BD,E0,20,F0,3,4C,6D,98
23 DATAA9,61,8D,B7,BD
24 DATAAE,D4,BD,8E,74,BE,A2,20,8E,D4,BD
25 DATAA2,1F,A0,00,BD,55,BE,A8,C0,64,F0,
8,CA,E0,00
26 DATAD0,F3,4C,9A,98
27 DATAA9,20,9D,55,BE,98,CA,9D,55,BE,CA,
4C,85,98
28 DATAAE,72,BE,E0,20,F0,3,4C,A9,98
30 DATAA9,61,8D,72,BE
31 DATAAE,55,BE,8E,F5,BE,A2,20,8E,55,BE
32 DATAA2,00,A0,00,BD,F5,BE,A8,C0,64,F0,
8
33 DATAE8,E0,1F,D0,F3
34 DATA4C,D6,98
35 DATAA9,20,9D,F5,BE,98,E8,9D,F5,BE,E8,
4C,C1,98
36 DATAAE,F7,BE,E0,20,F0,3
37 DATA4C,E5,98
38 DATAA9,61,8D,F7,BE
39 DATAAE,14,BF,8E,B4,BF,A2,20,8E,14,BF
40 DATAA2,1F,A0,00,BD,95,BF,A8,C0,64,F0,
8,CA,E0,00
41 DATAD0,F3,4C,12,99
42 DATAA9,20,9D,95,BF,98,CA,9D,95,BF,CA,
4C,FD,98
43 DATAAE,B2,BF,E0,20,F0,3,4C,01,99
45 DATAA9,61,8D,B2,BF
46 DATAAE,FB,BC,E0,20,F0,3,4C,30,99
48 DATAA9,61,8D,FB,BC
49 DATAAE,8F,BD,E0,20,F0,3,4C,3F,99
51 DATAA9,61,8D,8F,BD
52 DATAAE,4A,BE,E0,20,F0,3,4C,4E,99
54 DATAA9,61,8D,4A,BE
55 DATAAE,CF,BE,E0,20,F0,3,4C,5D,99
57 DATAA9,61,8D,CF,BE
58 DATAAE,8A,BF,E0,20,F0,3,4C,6C,99
60 DATAA9,61,8D,8A,BF,60

```

```
70 REM
71 DATA33,33,51,45,33,33,51,45
72 DATA08,08,62,08,28,22,16,48
73 DATA08,08,62,08,28,52,04,06
74 DATA08,28,62,63,61,59,30,12
75 DATA00,34,55,63,63,63,57,16
76 DATA08,61,63,63,63,63,29,24
77 DATA63,63,55,63,63,63,57,16
78 DATA08,61,63,63,63,63,63,63
79 DATA00,01,11,15,31,15,05,01
80 DATA00,00,32,52,60,62,56,16
81 DATA08,08,62,08,28,20,20,54
82 DATA08,42,60,08,12,18,49,01
83 DATA63,63,63,63,62,~4,08,00
84 DATA00,04,45,63,63,63,63,63
85 DATA63,63,63,63,63,63,63,63
86 DATA63,62,60,60,48,56,48,32
87 DATA63,62,56,48,48,56,48,60
88 DATA56,60,60,48,48,48,32,32
89 DATA32,32,32,48,56,56,56,60
90 DATA15,07,07,07,03,01,01,01
91 DATA01,01,03,03,03,15,15,07
92 DATA15,03,07,03,03,07,31,63
100 FORA=0T0364:READD$:D=VAL("#"+D$):POK
E#9800+A,D:CS=CS+D:NEXT
105 IFCS<>51891THENPRINT"MACHINE CODE ER
ROR":~ND
110 FORA=46856T047031:READD:POKEA,D:NEXT
115 CLS:PAPER4:INK7
120 GOSUB6000
190 X=6:Y=25:D=0:L=INT(RND(1)*6)+1
200 CLS:PAPER0:INK3
201 IFL=7THEN4000
202 IFL=-1THEN3000
203 FORA=0T026:PLOT0,A,L+1:NEXT
204 PLOT18,7,"a":PLOT6,11,"a":PLOT33,15,
"a":PLOT6,19,"a":PLO~33,23,
"a"
205 PLOT1,6," ieh e f e f e f e f e
f e f e f e f e J
206 PLOT1,10," iae f e f e f e f e f e g e f e f e f e
e f e f e f e f e J
207 PLOT1,14," i f e f e g Ø f e f e f e f e f e f e ~ e f e
f e f e f e faJ
208 PLOT1,18," iae f e f e f e f e f e f e f e f e f e
e f e f e fg feJ
209 PLOT1,22," i f e f e g e f e f e f e f e f e f e f e
e f e f e faJ
210 PLOT1,26," i f e f e f e f e f e f e f e f e f e f e
e f e f e fg feJ
220 CALL#9800
221 PLOT1,19,"o o q":PLOT1,20,"o o r":PLOT1,
21,"o o s":PLOT1,22,"m m m"
222 PLOT1,18,"o o o e f":
225 PLOT1,17,"o o s":PLOT1,16,"o o r":PLOT1,
15,"o o q":PLOT1,14,"o o o e f g f
```

PITFALL

```

230 PLOT1,13,"oos":PLOT1,12,"oor":PLOT1,
11,"o~q":PLOT1,10,"ooeafaf
235 PLOT1,9,"oos":PLOT1,8,"oor":PLOT1,7,
"ooq":PLOT1,6,"ooeefefefefe
feefefa
239 PLOT1,5,"oor
240 PLOT1,4,"oTRUEs
241 PLOT1,0,"oooooooooooooooooooooooooooo
ooooooooooooo
242 PLOT1,1,"oooooooooooooooooooooooooooo
ooooooooooooo
243 PLOT1,2,"oommmmmmmmmmmmmmmmmmmmmmmmmmm
mmmmmmmmmmmm
244 PLOT1,3,"oq
250 FORA=6T022STEP4
251 PLOT34,A,"efoo"
252 PLOT34,A+1," to
253 PLOT34,A+2," uo
254 PLOT34,A+3," oo
255 NEXT
256 PLOT34,26,"efoo
280 PLOT19,1," INJURIES"+STR$(D)+CHR$(L+
1)
290 PLOT2,1," LEVEL"+STR$(L)+CHR$(L+1)
300 IFX=SANDY=25THENL=L-1:X=32:Y=5:GOTO2
00
301 IFX=33ANDY=5THENL=L+1:X=6:Y=25:GOTO2
00
302 IFSCRN(X+1,Y)=100THEN1000
303 IFSCRN(X-1,Y)=10~THEN1000
305 CALL#9800:PLOTX,Y,A$
306 Q=INT(RND(1)*35):IFQ=9THENPLOT25,9,"
d":SOUND1,999,0:PLAY7,0,1,2
000
320 R=PEEK(520):IFR=56THEN302
321 IFR=172THENGOTO600'LEFT
322 IFR=188THENGOTO0500'RIGHT
323 IFR=156THENGOTO700'UP
324 IFR=180THENGOTO800'DOWN
325 IFR=132THENGOTO900'JUMP
380 CALL#9800
385 CALL#9800
390 GOTO320
500 IFSCRN(X+1,Y)=100THEN1000
510 PLOTX,Y," ":X=X+1
515 CALL#9500:SOUND1,X*3+9,6
520 IFX>34THENX=34
525 PLOTX,Y,"c":SOUND1,X,0
530 A$="c":GOTO300
600 IFSCRN(X-1,Y)=100THEN1000
610 PLOTX,Y," ":X=X-1
615 CALL#9500:SOUND1,X*3+9,6
620 IFX<5THENX=5
625 PLOTX,Y,"b":SOUND1,X,0
630 A$="b":GOTO300
700 IFSCRN(X,Y-1)=32THENGOTO300

```

```

710 FORP=YTOY-4STEP-1:IFSCRN(X,P-1)=100T
HEN1000
715 CALL#9500:SOUND1,P*9,7
720 PLOTX,P,"k":CALL#9800:WAIT5:PLOTX,P,
"a":NEXT:Y=P+1:PLOTX,Y,"k "
725 SOUND1,P,0
730 GOTO300
800 REM
801 IFY=25THEN300
802 IFSCRN(X,Y+3)=32THENGOTO300
805 PLOTX,Y," "
810 FORP=Y+1TOY+4:IFSCRN(X,P+1)=100THEN1
000
815 CALL#9500:SOUND1,P*9,7
820 PLOTX,P,"k":CALL#9800:WAIT3:PLOTX,P,
"a":NEXT:Y=P-1:PLOTX,Y,"k "
825 SOUND1,P,0
830 GOTO300
900 IFX<60RX>33THENGOTO300
904 FORP=YTOY-2STEP-1
905 SOUND1,(P*2),6
910 PLOTX,P,"l":CALL#9800:WAIT3:PLOTX,P,
" ":NEXT:Y=P+1
915 IFA$="b"THENX=X-1ELSEX=X+1
920 FORP=YTOY+2
925 SOUND1,(P*2),6
930 PLOTX,P,"l":CALL#9800:WAIT3:PLOTX,P,
" ":NEXT:Y=P-1
940 PLOTX,Y,"k
945 SOUND1,99,0
950 GOTO300
1000 PING:PLOTX-1,Y," "
1020 D=D+1:PLOT19,1," INJURIES"+STR$(D)+
CHR$(L+1)
1025 IFD=4THENGOTO2000
1030 GOTO300
2000 REM DEAD
2020 CLS:PRINT:PRINT:PRINT:PRINT:PRINT:P
RINT
2030 PRINT,,"YOU DIE FROM YOUR INJURIES.
2035 WAIT200
2040 FORA=1000TO1200STEP10
2050 SOUND1,A,0:SOUND2,A+2,0
2060 PLAY7,0,1,4000:WAIT1:NEXT
2070 GOTO5000
3000 CLS:FORA=2T025:PLOT1,A,"oq":PLOT36,
A,"oo":NEXT:INK1
3002 PLOT1,2,"oommmmmmmmmmmmmmmmmmmmmmmmm
mmmm
3005 PLOT1,26,"oohhhhhhhhhhhhhhhhhhhhhhhhh
hhhhhhhhhhoo
3010 FORA=2T025:PLOT34,A,"l":PLOT34,A-1,
" ":WAIT5
3020 SOUND1,A*10,0:SOUND2,A*11,0:PLAY7,0
,1,1000
3030 NEXT:PLOT34,25," ":EXPLODE

```

PITFALL

```

3040 GOTO5000
4000 CLS:PAPER4:IN~2
4010 FORA=20TO26:PLOT1,A,"oooooooooooooooooooo
oooooooooooooooooooooooooooo":NE
XT
4020 ~ORA=1TO38:PLOTA,19,"c":PL~TA-1,19,
" ":WAIT3:SOUND1,A*2,9:NEXT
4030 PLOT38,19," ":PING
4040 GOTO5000
5000 CLS:PAPER4:INK2
5010 PLOT10,10,CHR$(10)+"ANOTHER GO ?
5012 PLOT10,09,CHR$(10)+"ANOTHER GO ?
5015 PLOT14,12,"{(Y-N)

```

```

5020 GETT$
5030 IFT$="Y"THENRUN190
5040 IFT$="N"THENEND
5050 GOTO5020
6000 REM INSTRUCTIONS
6010 PLOT4,6,"LEFT ARROW.....LEFT
6020 PLOT4,8,"RIGHT ARROW.....RIGHT
6030 PLOT4,10,"DOWN ARROW.....DOWN
6040 PLOT4,12,"UP ARROW..~.....UP
6050 PLOT4,14,"SPACE BAR.....JUMP
6060 PLOT5,24,CHR$(12)+"PRESS ANY KEY TO
START
6070 GETA$:RETURN

```

FOOTBALL POOLS

```

1 REM FOOTBALL POOLS
2 REM
3 REM E HOLLISTER
4 REM
5 REM FOR ORIC-1 (USES 16K OF MEMORY)
6 REM
10 IF PEEK(#0400) <>3 THEN GOTO 2740
20 REM          DIVISION 1
30 DATA ARSENAL
40 DATA ASTON VILLA
50 DATA CHELSEA
60 DATA COVENTRY
70 DATA EVERTON
80 DATA IPSWICH
90 DATA LEICESTER
100 DATA LIVERPOOL
110 DATA LUTON
120 DATA MAN UTD
130 DATA NEWCASTLE
140 DATA NORWICH
150 DATA NOTTS FOREST
160 DATA Q.P.R.
170 DATA SHEFFIELD W
180 DATA SOUTHAMPTON
190 DATA STOKE
200 DATA SUNDERLAND
210 DATA TOTTENHAM
220 DATA WATFORD
230 DATA W.B.A.
240 DATA WEST HAM
250 REM          DIVISION 2
260 DATA BARNLEY
270 DATA BIRMINGHAM
280 DATA BLACKBURN
290 DATA BRIGHTON
300 DATA CARDIFF
310 DATA CARLISLE
320 DATA CHARLTON
330 DATA CRYSTAL PLC

```

```

340 DATA FULHAM
350 DATA GRIMSBY
360 DATA HUDDERSFIELD
370 DATA LEEDS
380 DATA MAN CITY
390 DATA MIDDLESBORO
400 DATA NOTTS C
410 DATA OLDHAM
420 DATA OXFORD
430 DATA PORTSMOUTH
440 DATA SHEFFIELD U
450 DATA SHREWSBURY
460 DATA WIMBLEDON
470 DATA WOLVES
480 REM          DIVISION 3
490 DATA BOLTON
500 DATA BOURNEMOUTH
510 DATA BRADFORD
520 DATA BRENTFORD
530 DATA BRISTOL CTY
540 DATA BURNLEY
550 DATA BRISTOL ROV
560 DATA CAMBRIDGE
570 DATA DERBY
580 DATA DONCASTER
590 DATA GILLINGHAM
600 DATA HULL
610 DATA LINCOLN
620 DATA NEWPORT
630 DATA MILLWALL
640 DATA ORIENT
650 DATA PLYMOUTH
660 DATA PRESTON
670 DATA READING
680 DATA ROTHERHAM
690 DATA SWANSEA
700 DATA WALSALL
710 DATA WIGAN
720 DATA YORK

```

FOOTBALL POOLS

```

730 REM                DIVISION 4
740 DATA ALDERSHOT
750 DATA BLACKPOOL
760 DATA BURY
770 DATA CHESTER
780 DATA CHESTERFIELD
790 DATA COLCHESTER
800 DATA CREWE
810 DATA DARLINGTON
820 DATA EXETER
830 DATA HALIFAX
840 DATA HARTLEPOOL
850 DATA HEREFORD
860 DATA MANSFIELD
870 DATA NORTHAMPTON
880 DATA PETERBOROUGH
890 DATA PORT VALE
900 DATA ROCHDALE
910 DATA SCUNTHORPE
920 DATA SOUTHEND
930 DATA STOCKPORT
940 DATA SWINDON
950 DATA TORQUAY
960 DATA TRANMERE
970 DATA WREXHAM
980 DIMH(92):DIMA(92)
990 DIMN(92):DIMDA$(92):DIMRA$(92)
1000 FORD=1TO92
1010 READDA$(D)
1020 NEXTD
1030 FORN=1TO92:N(N)=PEEK(#0400+N):NEXTN
1040 CLS:PRINT
1050 PRINT"-----MENU-----"
1060 PRINT
1070 PRINT
1080 PRINT"1 UPDATING THE FORM DATA"
1090 PRINT
1100 PRINT"2 OBTAINING THE FORECASTS"
1110 PRINT
1120 PRINT"3 SAVING UPDATED FORM"
1130 PRINT
1140 PRINT"4 END PROGRAM AND NOTES ON RE
LOADING"
1150 PRINT
1160 PRINT:INPUT"CHOICE 1,2,3,4";M$
1170 M=VAL(M$)
1180 IFM>4 ORM<1 ORM<>INT(M) THEN PING:G
OTO1040
1190 ON M GOSUB1210,1600,2370,2740
1200 GOTO 1040
1210 CLS:PRINT
1220 PRINT"        UPDATING THE FORM"
1230 PRINT
1240 PRINT"WHICH DIVISION NEEDS UPDATE"
1250 PRINT
1260 PRINT"CHOICE 1,2,3,OR 4

```

```

1270 PRINT
1280 INPUT"DIVISION";D$
1290 D=VAL(D$)
1300 IFD<1 ORD>4 ORD<>INT(D) THEN PING:G
OTO1210
1310 GOSUB 2590
1320 CLS:PRINT
1330 PRINT"UPDATING FORM  DIVISION "D
1340 PRINT
1350 PRINT"HL=HOME LOSE  AL=AWAY LOSE"
1360 PRINT"HD=HOME DRAW  AD=AWAY DRAW"
1370 PRINT"HW=HOME WIN   AW=AWAY WIN"
1380 PRINT"N=NO UPDATE REQUIRED"
1390 PRINT
1400 PRINT"ENTER LAST RESULT.HL,HD,HW,AL
,AD,AW,N"
1410 GOSUB2650
1420 FORT=S TO F
1430 PRINTDA$(T);:INPUT R$
1440 PRINT
1450 J=0
1460 IFR$="HL"THENR=0:J=1
1470 IFR$="HD"THENR=1:J=1
1480 IFR$="HW"THENR=3:J=1
1490 IFR$="AL"THENR=1:J=1
1500 IFR$="AD"THENR=4:J=1
1510 IFR$="AW"THENR=5:J=1
1520 IFR$="N"THENR=0:J=1
1530 IFJ=0THENPING:GOTO1430
1540 N(T)=R+PEEK(#0400+T)
1550 POKE#0400+T,N(T)
1560 NEXTT
1570 PRINT
1580 PRINT"END OF DIVISION "D
1590 PRINT
1600 INPUT"ANY MORE UPDATING Y/N";A$
1610 IFA$<>"Y"AND A$<>"N"THEN GOTO1600
1620 GOSUB2690
1630 IF A$="Y"THEN GOTO 1210
1640 RETURN
1650 CLS:PRINT
1660 PRINT"  OBTAINING THE FORECASTS"
1670 PRINT
1680 INPUT"DIVISION 1,2,3,OR 4 ";D$
1690 D=VAL(D$)
1700 IFD<1 ORD>4 ORD<>INT(D) THEN PING:G
OTO1650
1710 CLS:PRINT
1720 PRINT"FORECASTS FOR DIVISION "D
1730 PRINT
1740 GOSUB2590
1750 FORG=STOFSTEP2
1760 PRINTSPC(17);G+1;SPC(1);DA$(G+1)
1770 PRINTCHR$(11);G;SPC(1);DA$(G)
1780 NEXTG
1790 PRINT

```


FOOTBALL POOLS

```

1800 INPUT"HOW MANY GAMES IN THIS DIVISI
ON";B$
1810 IFASC(B$)=48 THEN GOTO1650
1820 B=VAL(B$)
1830 IFB>(F+1-S)/2 ORB<>INT(B) ORB<1 THE
N PING:GOTO2720
1840 GOSUB2650
1850 FORU=1TOB
1860 PRINT"MATCH NUMBER ";U
1870 PRINT
1880 INPUT"HOME TEAM NUMBER";H$
1890 H=VAL(H$)
1900 IFH<S ORH>F THENPING:GOTO1880ELSEPR
INTSPC(25)CHR$(11)DA$(H)
1910 PRINT
1920 INPUT"AWAY TEAM NUMBER";A$
1930 A=VAL(A$)
1940 IFA<S ORA>F THENPING:GOTO1920ELSEPR
INTSPC(25)CHR$(11)DA$(A)
1950 IFA=HTHENPING:GOTO1920
1960 H(U)=H
1970 A(U)=A
1980 W=N(H)-75
1990 X=INT(W/10)
2000 Y=W+X
2010 Z=W+X+X+X+X
2020 C=N(A)-75
2030 IFC<Y ORC=Y THENRA$(U)~"1"
2040 IFC>Z ORC=Z THENRA$(U)~"2"
2050 IFC>YANDC<Z THENRA$(U)~"X"
2060 PRINT
2070 NEXTU
2080 PRINT
2090 PRINT"TYPE ANY KEY FOR PRINTER INST
RUCTIONS AND FORECASTS FOR
DIV "D
2100 GETA$
2110 GOSUB2690
2120 PRINT
2130 PRINT"TURN ON PRINTER IF WANTED"
2140 PRINT
2150 INPUT"OUTPUT TO PRINTER Y/N";P$
2160 IFP$<>"Y"ANDP$<>"N"THENGOTO2150
2170 PRINT:PRINT
2180 PRINT"FORECASTS FOR DIVISION "D
2190 PRINT
2200 FORQ=1TOB
2210 K=13-LEN(DA$(H(Q)))
2220 J=13-LEN(DA$(A(Q)))
2230 PRINTDA$(H(Q));SPC(K);"V ";DA$(A(Q
));SPC(J);RA$(Q)
2240 IFP$="N"THEN GOTO 2260
2250 LPRINTDA$(H(Q));SPC(K);"V ";DA$(A(
Q));SPC(J);RA$(Q)
2260 NEXTQ
2270 PRINT:PRINT

```

```

2280 INPUT"ANY MORE FORECASTS Y/N";Q$
2290 IF Q$="Y" THEN GOTO 1650
2300 IFQ$<>"N" THENGOTO2280
2310 PRINT
2320 PRINT"END OF FORECASTING"
2330 PRINT
2340 PRINT"PRESS ANY KEY FOR MENU"
2350 GETA$
2360 RETURN
2370 CLS:PRINT
2380 PRINT"SAVING UPDATED FORM DATA"
2390 PRINT
2400 PRINT"1 PLACE CASSETTE"
2410 PRINT" INTO TAPE RECORDER"
2420 PRINT
2430 PRINT"2 SET COUNTER TO REQUIRED PO
SITION"
2440 PRINT
2450 PRINT"3 DEPRESS PLAY AND RECORD"
2460 PRINT
2470 PRINT"4 PRESS ANY KEY TO START SAV
ING"
2480 PRINT
2490 PRINT"5 SAVED AT SLOW SPEED"
2500 GETA$
2510 POKE#0400,3
2520 CSAVE" ",S,A#0400,E#0450
2530 PRINT:PRINT:PRINT:PRINT
2540 PRINT"UPDATED PROGRAM SAVED"
2550 PRINT
2560 PRINT"PRESS ANY KEY FOR MENU"
2570 GETA$
2580 RETURN
2590 IFD=1 THEN S=1:F=22
2600 IFD=2 THEN S=23:F=44
2610 IFD=3 THEN S=45:F=68
2620 IFD=4 THEN S=69:F=92
2630 IFD>4 THEN PING
2640 RETURN
2650 DOKE621,48680
2660 POKE623,9
2670 FORL=1TO9:PRINTCHR$(11)+CHR$(11):NE
XTL
2680 RETURN
2690 DOKE621,48000
2700 POKE623,27:CLS
2710 RETURN
2720 PRINTCHR$(11);SPC(38)
2730 PRINTCHR$(11)+CHR$(11):GOTO1800
2740 CLS:PRINT
2750 PRINT"PROGRAM TERMINATED"
2760 PRINT:PRINT:PRINT
2770 PRINT" NOTES ON RELOADING"
2780 PRINT
2790 PRINT"1 LOAD FORM DATA FILE"
2800 PRINT

```

FOOTBALL POOLS

```
2810 PRINT"2 TYPE NEW AND RETURN"
2820 PRINT
2830 PRINT"3 LOAD MAIN PROGRAM"
2840 PRINT
2850 PRINT"4 LOAD AT SLOW SPEED"
```

```
2860 PRINT
2870 PRINT"5 TO RUN WITH NO FORM DATA TA
PE LOADED TYPE RUN20 (FOR T
ESTING)"
2880 END
```

ETCH-A-SKETCH

```
10 REM      ETCH-A-SKETCH
15 REM
20 REM      BY
25 REM
30 REM      PAT COOPER
35 REM
40 REM      OCT.1984
45 REM
50 HIMEM#4FFF
60 GOSUB10000
100 ON MODE+1 GOSUB9000,2000,2000,200
0,3000,4000,5000,6000,7000,8000
110 GOSUB1000'GET KEY
120 GOTO100
1000 K$=KEY$
1010 X=X+(K$=",")+(K$="K")+(K$="O")-(
K$="/" )-(K$="'" )-(K$="[" )
1015 IFX<0THENX=0ELSEIFX>239THENX=239
1020 Y=Y-(K$=",")-(K$="." )-(K$="/" )+(
K$="O" )+(K$="P" )+(K$="[" )
1030 IFY<0THENY=0ELSEIFY>199THENY=199
1040 IFK$=" " THEN GOSUB1500
1050 IFK$=CHR$(127) THEN HIRES:MODE=3:
POKE#26A,10:RETURN
1060 IFK$=CHR$(27) THEN GOSUB1800
1070 IF K$="\ " THEN CALL#51A0:RETURN
1080 IFK$="-" THEN CALL#50B0:RETURN
1090 IF K$="=" THEN CALL#51E2:CLS:RET
URN
1180 IFK$<"0"ORK$>"9" THEN RETURN
1190 OLDMODE=MODE:MODE=VAL(K$)
1199 RETURN
1500 CURSETX,Y,FB
1510 DRAWX1-X,Y1-Y,FB
1530 CURSETX,Y,FB
1540 DRAWX2-X,Y2-Y,FB
1599 RETURN
1800 X2=X1:X1=X:Y2=Y1:Y1=Y
1899 RETURN
2000 CURSETX,Y,(MODEAND3)-1
2010 GOSUB2500'DETAILS TO SCREEN
2090 CURSETX,Y,(MODEAND3)-1
2099 RETURN
2500 PRINTCHR$(30)"  MODE
      X POSN  Y POSN"
```

```
2520 PRINT:PRINTMODE$(MODE),,X,Y;
2599 RETURN
3000 CURSETX,Y,3:DRAWX1-X,Y1-Y,2
3010 CURSETX,Y,2:DRAWX2-X,Y2-Y,2
3020 GOSUB2500
3030 CURSETX,Y,3:DRAWX1-X,Y1-Y,2
3040 CURSETX,Y,2:DRAWX2-X,Y2-Y,2
3090 FB=1
3099 RETURN
4000 CURSETX,Y,3:DRAWX1-X,Y1-Y,2
4010 CURSETX,Y,2:DRAWX2-X,Y2-Y,2
4020 GOSUB2500
4030 CURSETX,Y,3:DRAWX1-X,Y1-Y,2
4040 CURSETX,Y,2:DRAWX2-X,Y2-Y,2
4090 FB=0
4099 RETURN
5000 GOSUB2500'DETAILS TO SCRIN
5010 BASE=0
5030 GETK$
5040 IFK$="6" THEN BASE=128:GETK$
5050 IFK$<"@ "ORK$>"W" THEN 5010
5060 CURSETX,Y,3
5070 I=ASC(K$)-64
5080 FILL1,1,I+BASE
5090 MODE=OLDMODE
5099 RETURN
6000 GOSUB2500'DETAILS TO SCRIN
6010 POKE#E,X:POKE#F,Y
6020 CALL#5000 'FILL
6050 MODE=OLDMODE
6060 PING
6099 RETURN
7000 GOSUB2500'DETAILS TO SCRIN
7010 GETK$:IFK$<"0" OR K$>"9" THEN 70
10
7020 R=VAL(K$)
7030 GETK$:IFK$<"0" OR K$>"9" THEN 70
30
7040 R=R*10+VAL(K$):IFR=0 THEN 7090
7050 IF X+R>239 OR X-R<0 THEN 7500
7060 IF Y+R>199 OR Y-R<0 THEN 7500
7070 CURSETX,Y,3
7080 CIRCLE R,1
7090 MODE=OLDMODE
7099 RETURN
```

ETCH-A-SKETCH

```

7500 FORI=0TO2*PISTEP 1/R
7510 XC=X+SIN(I)*R
7520 YC=Y-COS(I)*R
7530 IFXC<0THENXC=0ELSEIFXC>239THENXC
=239
7540 IFYC<0THENYC=0ELSEIFYC>199THENYC
=199
7550 CURSETXC,YC,1
7560 NEXT
7570 PING
7590 MODE=OLDMODE
7599 RETURN
8000 GOSUB2500'DETAILS TO SCREEN
8010 CSAVE"SKETCH DATA",A#A000,E#BF67
8020 PING
8030 MODE=3
8099 RETURN
9000 GOSUB2500'DETAILS TO SCREEN
9010 CLOAD"SKETCH DATA"
9020 PING
9030 MODE=3
9099 RETURN
10000 TEXT:MODE=3:POKE#26A,10
10010 X=119:Y=99:X1=119:Y1=99:X2=119:
Y2=99
10020 FORI=0TO9:READMO$(I):NEXT
10030 DATA" LOADING "," UNPLOT "," P
LOT "," MOVE "," DRAW "," ERASE
"
10040 DATA"COLOUR ?"," FILL ","CIRC
LE ?"," SAVING "
10050 GOSUB20000' INSTRUCTIONS.
10060 HIRES
10070 CALL#51E2
10080 HIRES
10090 POKE#26A,10
10099 RETURN
20000 READ CH
20010 REPEAT
20020 FOR I=0 TO 7
20030 READ CV:POKE#B400+CH*8+I,CV
20040 NEXT
20050 READ CH
20060 UNTIL CH=#FF
20100 DATA123,0,15,3,5,9,16,32,0
20110 DATA124,0,1,2,36,40,48,60,0
20120 DATA93,0,32,16,9,5,3,15,0
20130 DATA94,0,4,14,21,4,4,4,0
20140 DATA38,0,4,4,4,21,14,4,0
20150 DATA95,0,60,48,40,36,2,1,0
20160 DATA43,0,4,8,16,63,16,8,4
20170 DATA125,0,8,4,2,63,2,4,8
20180 DATA#FF
21000 FORI=#BB80 TO #BBA7
21010 POKEI,32
21020 NEXT

```

```

25000 CLS
25010 PAPER0:INK2
25020 PLOT10,10,"DO YOU WANT INSTRUCT
IONS ?"
25030 PLOT15,20,"PRESS 'Y' OR 'N'"
25040 GETK$
25050 IF K$="N" THEN 30500
25060 IF K$<>"Y" THEN 25040
30000 CLS:PAPER0:INK6:PRINTCHR$(4)
30010 PRINTTAB(13)CHR$(27)"JETCH-A-SK
ETCH."
30020 PRINT:PRINT:PRINT:PRINTCHR$(27)
"J Keys used to control cursor"
30030 PRINT:PRINTCHR$(27)"J direct
ion are :-":PRINT:PRINT
30040 PRINT:PRINTCHR$(27)"J
_ ^ ("
30050 PRINT:PRINTCHR$(27)"J
O P ["
30060 PRINT:PRINTCHR$(27)"J +
K ' )"
30070 PRINT:PRINTCHR$(27)"J
, . /"
30080 PRINT:PRINTCHR$(27)"J
! & "]"
30090 PRINT:PRINT:PRINT:PRINTCHR$(27)
"J Press any key to continue..."
30099 PRINTCHR$(4);:GETK$
30100 CLS:PRINTCHR$(4)
30110 PRINTTAB(13)CHR$(27)"JETCH-A-SK
ETCH.":PRINTCHR$(4):PRINT
30120 PRINT" Several keys are used t
o select the"
30130 PRINT" required mode as follow
s :-"
30140 PRINT:PRINT
30160 PRINT" KEY
MODE"
30170 PRINT" ---
----":PRINT
30200 PRINT" 1.....
UNPLOT"
30210 PRINT" 2.....
PLOT"
30220 PRINT" 3.....
MOVE"
30230 PRINT" 4.....
DRAW"
30240 PRINT" 5.....
ERASE"
30250 PRINT" 6.....
COLOUR"
30260 PRINT" 7.....
FILL"
30270 PRINT" 8.....
CIRCLE"

```

ETCH-A-SKETCH

```

30280 PRINT"          9.....
SAVE"
30290 PRINT"          0.....
LOAD"
30300 PRINT:PRINT:PRINT"      Press an
y key to continue..." :GETK$
30310 CLS:PRINTCHR$(4)
30320 PRINTTAB(13)CHR$(27)"JETCH-A-SK
ETCH." :PRINTCHR$(4):PRINT
30330 PRINT" Other keys produce the
required"
30340 PRINT" function as follows :-"
30350 PRINT:PRINTSPC(12)"MEMORY FUNCT
IONS"
30360 PRINTSPC(12)"-----"
30370 PRINT:PRINT"      -.....
.....COPY"
30380 PRINT:PRINT"      =.....
.....SWOP"
30390 PRINT:PRINT"      \.....
.....CLEAR"
30400 PRINT:PRINTSPC(12)"SCREEN FUNCT
IONS"
30410 PRINTSPC(12)"-----"
30420 PRINT:PRINT"      ESC.....
....MARK POINT"
30430 PRINT:PRINT"      SPACE.....
....FIX LINE"
30460 PRINT:PRINT"      DEL.....
....CLEAR SCREEN"
30470 PRINT:PRINT:PRINT"      Press an
y key to continue..." :GETK$
30500 IF PEEK(#5000)=#20 THEN RETURN
40000 DATA 209C5120B85120805020305020
0051A5
40010 DATA 0BC9C8D0F4202051EAEAEA2030
502020
40020 DATA 51A50BC9FFD0F4EAEAEA600000
000000
40030 DATA 206050A200204051A50AC9F0D0
F22058
40040 DATA 51206050A200205851A50AC9FF
D0F220
40050 DATA 40516000000000000000000000
000000
40060 DATA 207C51F00B207451D006208050
208451
40070 DATA 60000000000000000000000000
000000
40080 DATA 20DA50A2042040512090512058
512058
40090 DATA 51209051204051A206202D5120
905120
40100 DATA 0D51200D51209051202D516000
000000
40110 DATA 20D851A000B100910A88D0F9E6

```

```

01E60B
40120 DATA CA10F060000000000000000000
000000
40130 DATA 0000000000000000000000A203B5
009504
40140 DATA CA10F960A20DBDF0509500CA10
F86000
40150 DATA 00A00078000000000000FF000020
FF0000
40160 DATA A202200D51A200200D51E60B60
18B500
40170 DATA 69289500B50169009501600000
000000
40180 DATA A202202D51A200202D51C60B60
38B500
40190 DATA E9289500B501E9009501600000
000000
40200 DATA 460C900EF600F602D004F601F6
03A920
40210 DATA 850CE60A600000000060C240C50
12B500
40220 DATA 08D600D60228D004D601D603A9
01850C
40230 DATA C60A6000A000B100250C6000A0
00B102
40240 DATA 250C6000A000B104050C910460
000000
40250 DATA A000B106050C9106600000000A9
97D002
40260 DATA A977A21F8501A0008400A94091
0088D0
40270 DATA FBC601CA10F6600020E450A40F
F00620
40280 DATA 005188D0FAA200A40EF0062040
5188D0
40290 DATA FA20DA502090516020E450A958
850BA2
40300 DATA 1F6020D851A000B1009102B10A
9100B1
40310 DATA 02910A88D0F1E601E603E60BCA
10E660
40320 DATA
45000 TEXT:CLS:PLOT10,10,"TRANSFERING
M/CODE"
45010 PLOT13,15,"PLEASE WAIT !"
45020 A=#5000:READD$
45030 FORI=1TOLEN(D$)STEP2
45040 U=VAL("#"+MID$(D$,I,2)):POKE A,
U
45050 A=A+1:UT=UT+U
45060 NEXT
45070 READD$:IFD$<>"Z"THEN45030
45080 IFUT=40848 THEN RETURN
45090 PRINT:PRINT:PRINT"ERROR - CHECK
PROGRAM AGAINST LISTING." :PING
45100 END

```

EXTENSION ROM BOARD



If any other language or m/c code program (other than Oric Basic) is required, then it has to reside in RAM, i.e. use up valuable program space. It would be much more convenient if the required routines were in EPROM, which could be switched in to replace the Oric Basic ROM, when needed. The circuit described allows for just this (up to 4 ROMs plus Oric's internal ROM), so that the user can switch from one to the other, at will.

Design

The ROM board design was based upon the 2764 and 27128 EPROM chips, for which several programmer designs have already been published in various magazines.

A single 3 to 8 line decoder is used to do the address decoding and also the EPROM selection. This gives a theoretical maximum of eight EPROMs, but due to various reasons, it was decided to limit the design to four EPROMs. This still enables a choice of two 16K blocks (2764s) or four 16K blocks (27128s). A quad nand gate I.C. is used to provide various logic conditions to complete the circuit.

For anyone who is adventurous enough, the design may be extended to include extra EPROM sockets. This would entail address and data line buffering, plus the addition of a separate power supply. As this complicates the circuit, it is not described in any detail. If implemented, the full design would give you a massive 144K of switchable ROM!!

Also included is the often required hard reset button. This will be needed to restart the system, when switching between ROMs. Not only is this more convenient, but it saves wear and tear on the power plug.

One point that may cause problems for some users, is the

ROMDIS lead. This disables the Oric internal ROM, when taken low, but only works if mask PROMS are inside the machine (23128s). EPROMs were fitted to earlier Orics (Serial No less than 20,000), in which case, the PROMs will have to be removed and re-inserted in the extension EPROM Board.

How it works

Most of the work is done by the three to eight line decoder (IC2). This chip has six inputs, three of which determine which outputs are active, and the others being chip select (CE or \overline{CE}) Pins.

Consider the chip select inputs first . . .

- PIN 4 (\overline{CS}) Enables the chip when address lines 14 and 15 are high. ($\$C094$ ~~094~~ ~~094~~ to $\$FFF$). This is the area of memory that is used for the Oric ROM.
- PIN 5 (\overline{CS}) This pin is controlled by SW1, which selects either Oric's internal ROM or the external ROM. The signal that selects the external ROMs via IC2, is the same signal that disables the Oric ROM via the pin labelled ROMDIS
- PIN 6 (CS) Is logic high when Oric is reading from memory. (Logic low when writing to memory). When operating in ROM, the memory chip only needs to be enabled during the read cycle.

Therefore, IC2 will only be enabled under the following conditions . . .

1. Oric accessing memory between addresses $\$CDDD$ and $\$FFFF$.
2. External ROM board selected by SW1.
3. Oric is reading from memory.

The other three inputs select which ROM chip (IC3-6) is enabled.

- PIN 3 Is tied to OV because only four ROM chips are catered for in this design. If connected to an additional switch (SW5?) with a pull up resistor, then up to eight ROM chips can be used.

- PIN 1 Can be connected in one of two ways.
 1. Connect 1-2. Selects 1 of 4 ROMs. ROM type 27128.

Features

2. Connect 2-3. Selects 1 pair from 2 pairs of ROMs. A13 determines which ROM of the pair is active. ROM type 2764.

The \overline{OE} input to the ROMs is derived from the $\phi 2$ clock signal. IC1b being used as an inverter.

As in normal digital practice, all unused inputs are tied to definite logic levels to reduce transient switching noise. An abundance of decoupling capacitors should be distributed around the circuit, preferably as close as possible to IC power supply pins.

Switches

SW1-3

Min Slide

3 OFF

SW4

Push to make

1 OFF

Plugs & Sockets

34W IDC Plug

1 OFF

14W DIL IC Skt

1 OFF

16W DIL IC Skt

1 OFF

28W DIL IC Skt

4 OFF

Miscellaneous

34W IDC Ribbon Cable

Approx 4"

P.C.B.

Parts List

Resistors 2K2 2 OFF

Capacitors 100 μ F 1 OFF

100nf 5 OFF

Integrated Ccts

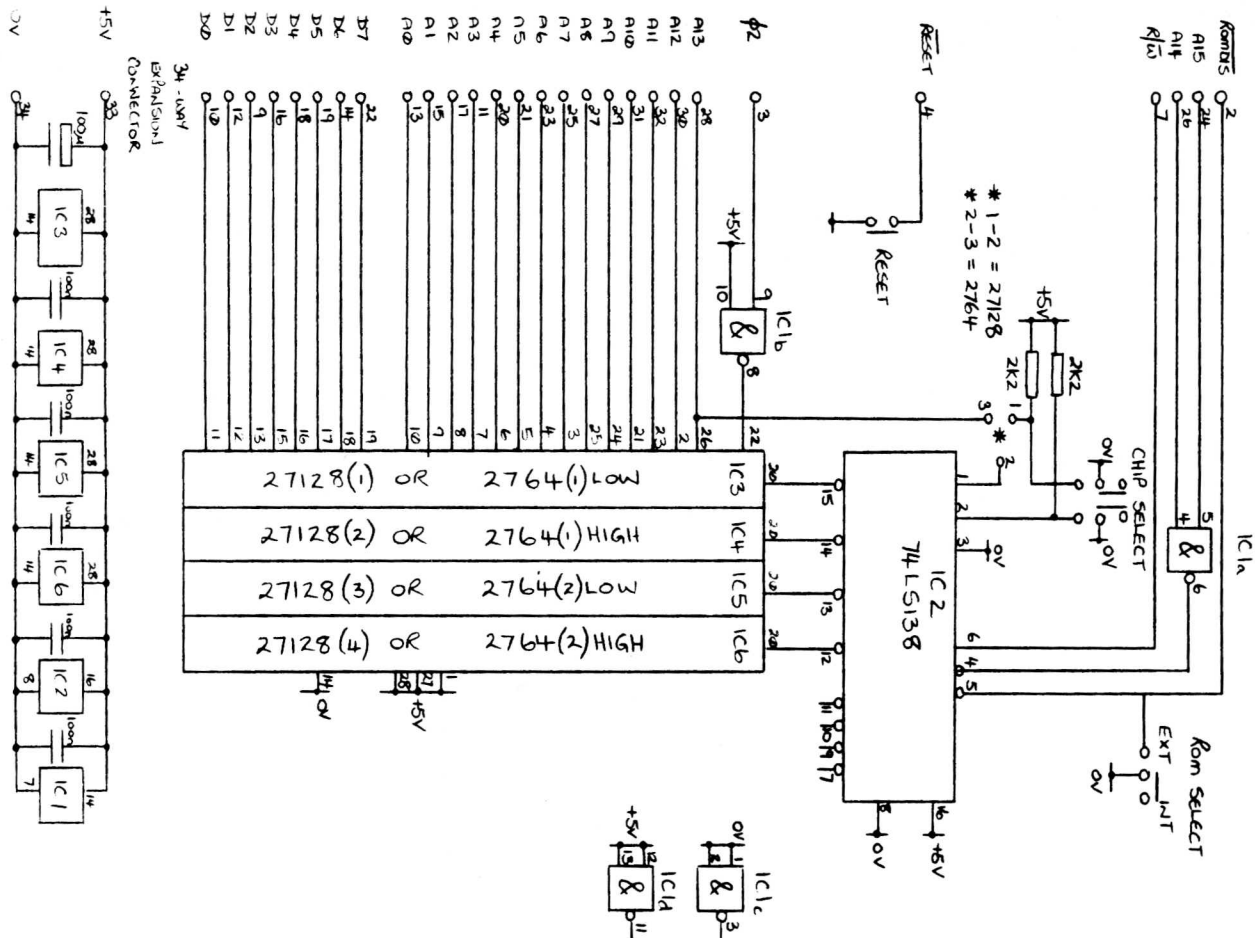
IC1 74LS00 1 OFF

IC2 74LS138 1 OFF

IC3-6 2764 or 27128 4 OFF *See text

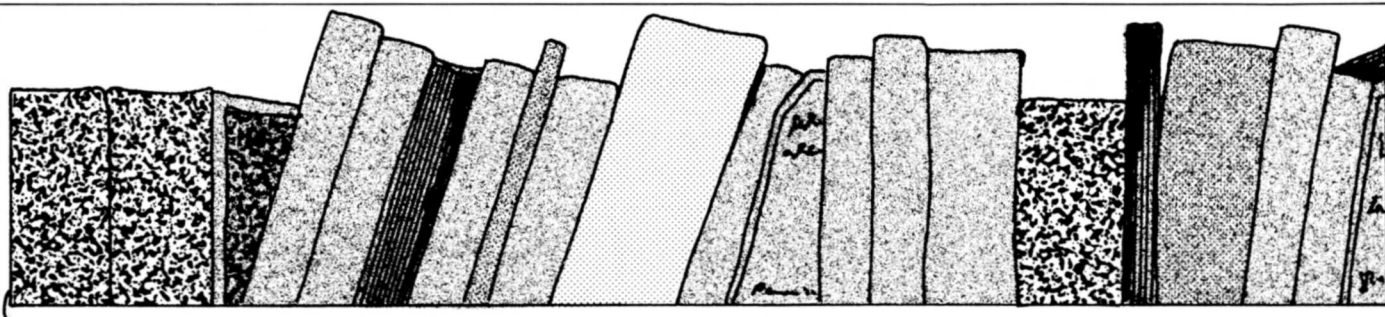
Buy Lines

Most of the parts should be easy to obtain from any mail order companies that advertise in electronics magazines. Suitable switches were obtained from R.S. Components Ltd.



by P. COOPER

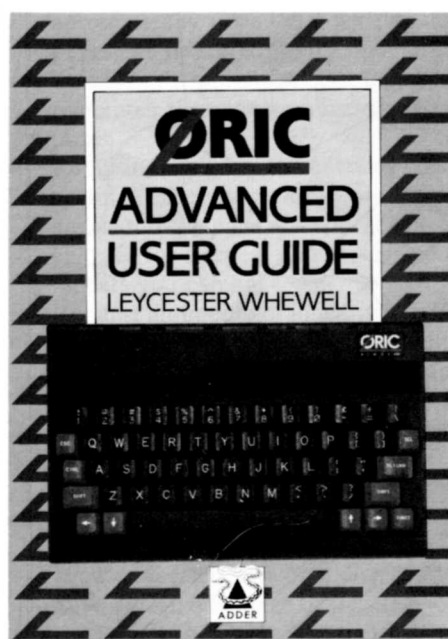
On The Bookshelf



Oric Advanced User Guide
Leycester Whewell
Adder Publishing
287pp £8.95

Computer Terminology Explained
I.D. Poole,
Babani Publishing Ltd.
81pp £1.95

Computers Mean Business
Jacquetta Megarry
Pan Breakthrough Books
336pp £2.95



This book is laid out alphabetically, like a dictionary, giving very brief explanations for some of the common terms encountered by the home computer enthusiast.

The amount of words listed seem to be very few, compared with the amount there actually are. The other thing is that the words listed are the ones that even people with very little knowledge of computers already know.

It has a table of ASCII and Control codes in the appendices, followed by four blank pages for notes?

Personally, I wouldn't recommend it. Only someone who knows nothing at all about computers and doesn't want to know much more, could consider it of any use. This may sound harsh but it is also rather overpriced, which doesn't help matters.

If you've been looking for a book that goes into more detail than the actual manual you get with your Atmos, this is it. This book really will be of great help to you and is well worth investing in.

It has a full ROM disassembly for anyone wanting to do Machine Code programming and full details of all the memory locations used by the ROM.

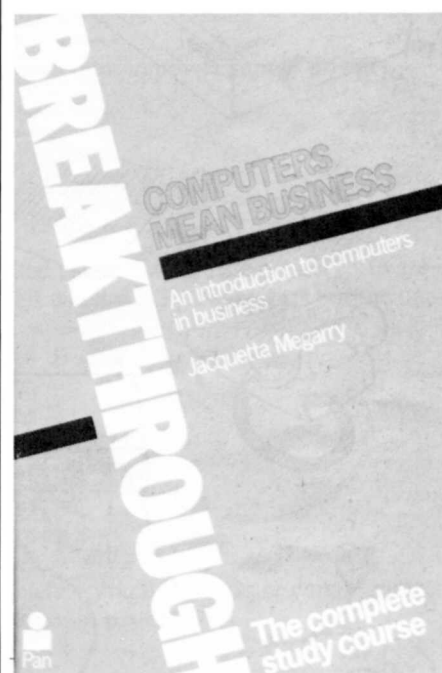
It explains how to read the keyboard in Machine Code and how to generate sound. There's a detailed explanation of how Basic works and Basic variable storage.

Apart from being much more comprehensive than any other book of its type, so far, it also covers Oric's hardware in great detail (mainly the Disc Drive).

I find it very impressive; I can confidently recommend it to anyone wanting a lot more information about their Oric-1/Atmos.

Computer Terminology Explained

I. D. POOLE



Although this book is not specifically for the Oric-1/Atmos it gives a detailed explanation of the key business applications of computers.

It includes Data Base management, Accounting, Stock control, Word Processing, choosing Hardware and Software, etc.

It is divided into four sections:

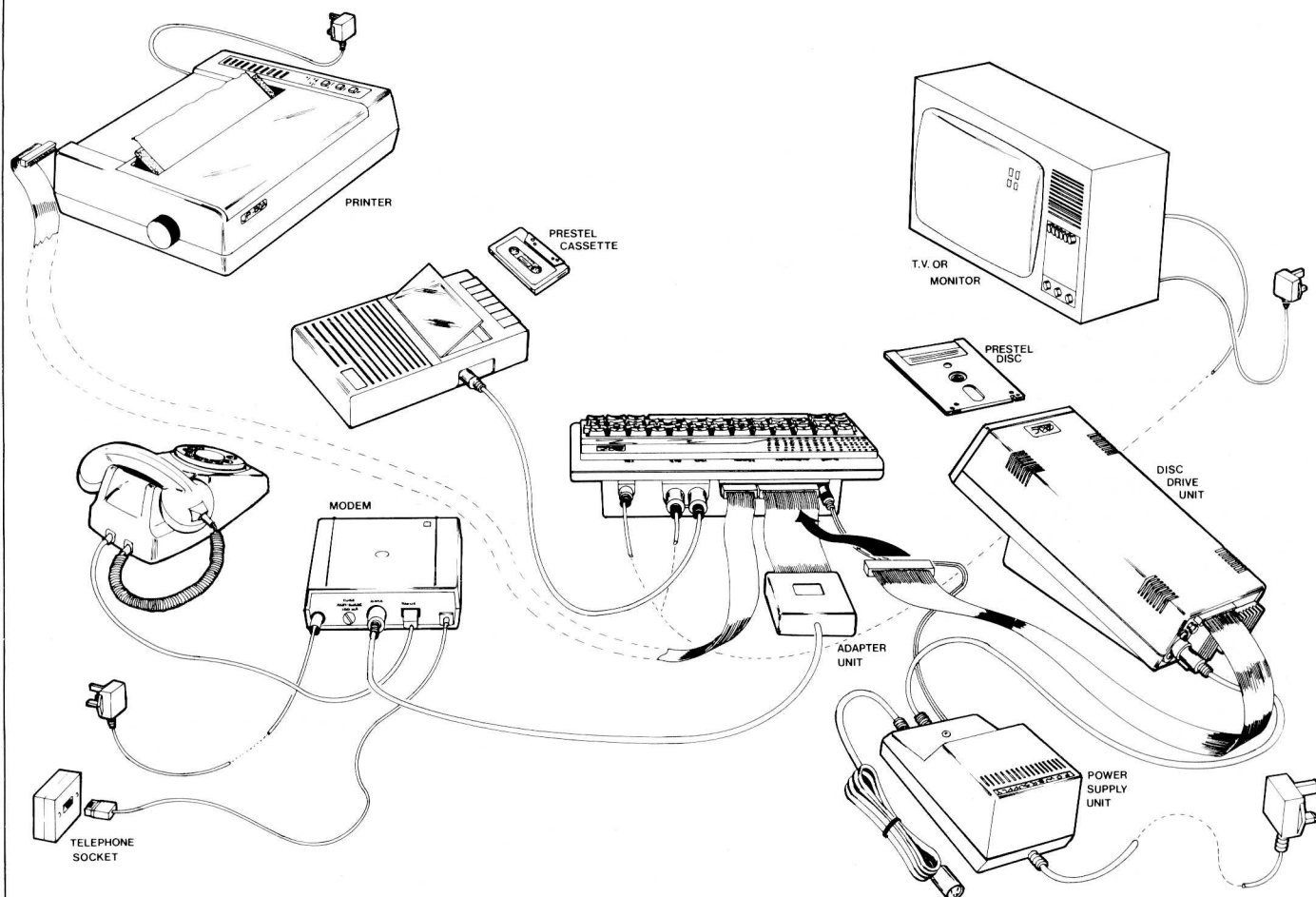
1. What is a computer?
2. How can computers help business?
3. Computers in business practice.
4. Computing and the organization.

There are two useful appendices. The first gives an alphabetical list of sources of advice and information, giving a brief description of the companies and how they can be of use to you. The second lists books for further reading with brief summaries of each of them.

At £2.95, it is good value, if you've an interest in business.

Features

THE ORIC MODEM



The ORIC Prestel system enables ORIC-1 or ATMOS owners to use their microcomputer as a fully functional Prestel terminal.

The system comprises a BT approved V23 modem, a Prestel terminal applications program cassette, a modem adapter unit and interconnecting cables.

The total cost of the Prestel system is £99.95 and this combined with the fact that a disc drive is not required, makes it a low cost system well within the reach of most ORIC-1 and ATMOS owners.

WHAT DO YOU NEED?

Besides owning an ORIC-1 or ATMOS and a tape recorder, you will need a BT type 600 plug-socket connecting your telephone to the telephone line. If your connection is the old type wired in junction box you will have to have it changed by British Telecom.

INSTALLATION

Installation is simple: Just unplug your telephone cable from the wall socket and connect the cable from the modem in its place, then plug the telephone cable into the socket at the rear of the modem. Next, connect the modem adapter to the microcomputer EXPANSION socket using the ribbon type cable, and to the modem using the DIN plug cable. No changes are necessary on your microcomputer so the entire procedure can be carried out in a few minutes. The connections from microcomputer to tape cassette recorder and T/V are unchanged.

Finally, insert the Prestel cassette into the tape recorder and load the program in the usual way (using CLOAD"). When loading is complete, an ORIC Prestel message is displayed and the microcomputer is now ready for use as a Prestel terminal.

THE ORIC MODEM

THE PRESTEL SYSTEM FACILITIES AVAILABLE

The following facilities are available to you. Most of these are 'on line', in other words your system must be connected to Prestel. Those facilities which do not require the Prestel connection are indicated by "off line".

- Select a Prestel page by number or by using the menu instructions displayed. When a page is selected, it is automatically stored in RAM; up to 99 average size pages can be stored.
- Re-select the previously displayed page without having to remember the page number, up to a maximum of three times.
- Request a repeat of the currently displayed page. This is useful in cases where some of the information displayed is garbled. If the page is a chargeable one, you won't be charged for a repeat and there is no limit to the number of repeats you can have.
- Request an update of information. Some pages are regularly updated (e.g. airline travel times). If you select such a page you will get the latest update. There is a charge each time you request an updated page.
- Select the next frame in alphabetic sequence under the currently displayed page number. E.g. if your screen is displaying page number 120 frame B, you can quickly select all subsequent frames (C, D etc) for that page number *only*.
- View concealed data on a displayed page.
- Save a displayed page or page(s) stored in computer memory onto tape cassette. To save a displayed page the command is CTRL+T. To save the pages currently stored in RAM, exit Prestel using CTRL+E and then go to the initial menu via CTRL+E again. You will then be given the option of saving to tape if you wish to do so.
- Print a displayed page. Use CTRL+L.
- Load a page from cassette tape, the page will be retained in RAM and displayed on the screen. Use the menu which is displayed immediately the Prestel software is loaded; it gives you the option of calling Prestel or retrieving a page from tape.
- Disconnect the system from Prestel but leave the ORIC Prestel software loaded. Use CTRL+E.
- Select a page for display and save it onto cassette tape in one operation.

(the maximum amount of data required to fill a screen including attributes and concealed characters). This leaves sufficient RAM (32K) to store ≈ 30 frames full of data ($40 \div 960$ bytes = approximately 38K bytes), or 99 frames with an average amount of data.

Modem

The modem is a V23 and is British Telecom approved, operating at 1200 baud in half duplex mode or 1200/75 baud in full duplex mode.

The modem has its own built in power supply driven by 240 volt AC mains.

Keyboard Characters

The following keys when pressed whilst the prestel software is loaded, do not give their usual ASCII characters but the ones indicated below:-

#	gives	£
[gives	←
	gives	$\frac{1}{2}$
]	gives	→
^	gives	↑
£	gives	#
	gives	$\frac{1}{4}$
	gives	$\frac{3}{4}$

DELETE: does not delete – you'll have to use cursor control to edit message and response pages.

Physical Dimensions

The units are quite small, the modem measures approximately 23.5cm ($9\frac{1}{4}$ " \div 16.5cm ($6\frac{1}{2}$ " \div 5.5cm ($2\frac{1}{4}$ " and the modem adapter measures 12cm ($4\frac{3}{4}$ " \div 6.5cm ($2\frac{1}{2}$ " \div 5cm (2").

TECHNICAL INFORMATION

Memory (RAM) Allocation

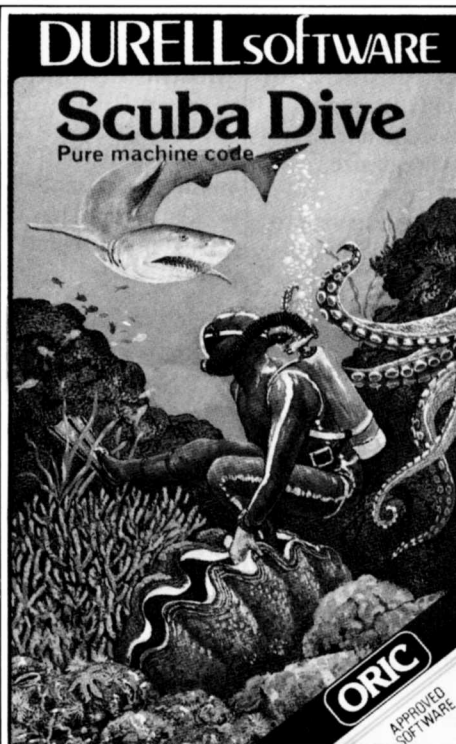
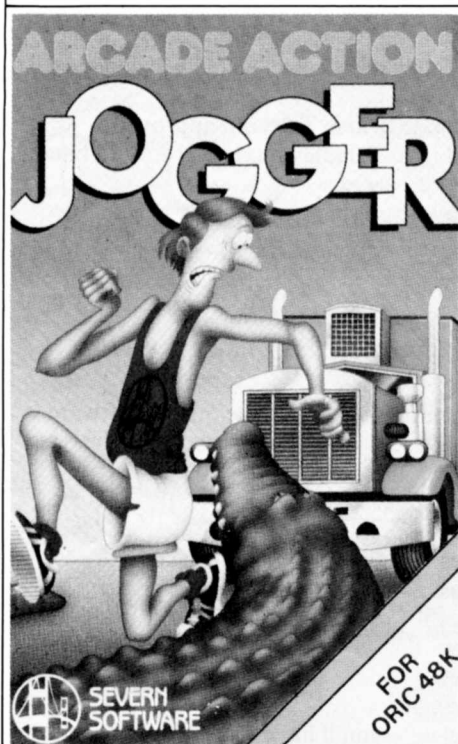
The Prestel software, when loaded into the computer memory occupies approximately 6K bytes of RAM within the bottom 48K (from 8000 to 9800) and a further 10K RAM is taken up with screen memory.

The area of RAM reserved for a page of data is 1096 bytes



OFFERS

Our opening offers come from Durell, Severn Software, Lothlorien, Micrograf and Haresoft. I'm sure you'll be able to find something that appeals to you.



Scuba Dive – Dive for sunken treasure, avoiding sharks, octopii and so on. For Oric-1 users, reduced from £6.95 **£2.95**

Hareraiser – Solve the puzzle on this and the follow up program and you might win a valuable prize. Save £1.00 **£7.95**

Oricaid – an excellent toolbox, monitor and assembler, from Micrograf and for all Orics. Save £2.00 on usual price **£9.95**

Galaxy 5 – A collection of five games set in space, from Durell. Suitable for 16K machines and reduced from £6.95 **£2.95**

Dinky Kong – Attempt to rescue the girl held captive by an angry gorilla. Including full colour graphics. Normally £6.95 **£2.95**

Atmos Extended Basic – Offers 22 extra commands, plus the facility of entering a line or command in upper or lower case. Save almost £2.00 **£7.00**

Super Meteors – A version of 'Deluxe-Asteroids' with excellent sound effects, hires graphics etc. Normally £6.95 **£4.50**

Jogger – Guide your intrepid Jogger across the four-lane motor-way amongst other things. Usually £6.95 **£2.95**

Galaxians – A superb version of the popular arcade game, written in super fast machine code. Was £6.95 **£4.50**

Lone Raider – You captain a spaceship and your mission is to save the Earth from aliens. Down from £8.50 **£7.00**

Acheron's Rage – You are enclosed by forcefields, trapping you in a ring of space. Normally £6.95 **£4.50**

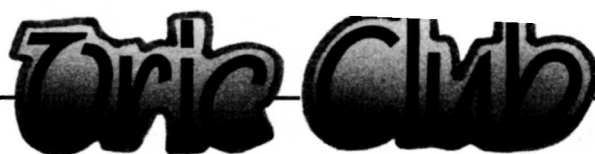
Orion – A two-pass symbolic assembler, disassembler with single-step monitor. Includes comprehensive manual. Was £12.95 **£7.95**

Johnny Reb – Play against the computer or a friend and perfect your tactics and rewrite American history. Save £3.00 **£3.95**

Dracula's Revenge – As fearless Friar Freddy, you receive the papal edict to sanctify all of Dracula's castles. Reduced from £6.95 **£4.50**

Two Gun Turtle – It's the middle of winter and as the turtle you must defend your strawberry patch. Save £4.00 **£2.95**

Warlord – A challenging game of analytical skill, set in 13th Century Japan. You must meet attacks from other armies. Normally £6.95 **£2.95**



OFFERS

Machine code for the Atmos and Oric 1 gives a full description of all the machine code instructions available to the Oric's 6502 chip and also suggests applications for their use. It describes how, when and where machine code routines can be entered. Talk to your Oric in its own language. **£5.95.**

Exploring Adventures on the Oric 48K gives plenty of suggestions and has three complete games listed, for you to play and study. Using the features of the Oric to good effect is well explained, and there are many hints for adventure game players too. With the normal price at £6.95, the club price of **£5.00** is a good saving.

The Altai joystick interface is fully compatible with the PASE standard interface, and works with any Oric computer. It takes one or two Atari compatible joysticks and plugs into the printer port, so may affect the sound of some games. Supporting software is supplied and could be incorporated into your own games very easily. The special club price is a mere **£4.95.**

A Hard Cover should prove very useful to those of you wanting to take care of your Oric-1/Atmos. It protects it from knocks, spillage and dust when not in use. It's made from tough but light ABS, shaped neatly to fit around your machine and it looks more attractive and offers more protection than ordinary soft covers. **£5.95.**

ORDERS

Item	Price	Number	Total
Software-Oric-1/Atmos			
Hareraiser	£7.95		
Oricaid	£9.95		
Atmos Extended Basic	£7.00		
Super Meteors	£4.50		
Galaxians	£4.50		
Lone Raider	£7.00		
Acheron's Rage	£4.50		
Orion	£7.95		
Dracula's Revenge	£4.50		
Two Gun Turtle	£2.95		
Software-Oric-1			
Scuba Dive	£2.95		
Galaxy 5	£2.95		
Dinky Kong	£2.95		
Jogger	£2.95		
Johnny Reb	£3.95		
Warlord	£2.95		
Books			
Machine Code for the Atmos and Oric-1	£5.95		
Exploring Adventures on the Oric 48K	£5.00		
Accessories			
Altai Joystick Interface	£4.95		
Hard Cover	£5.95		
Total			

Complete the order form and send it to:

Oric Club,
Unit 1,
Techno Park,
Newmarket Road,
Cambridge CB5 8PB

Cheques must be made payable to Oric Club. Alternatively, ring the Club on 02205 2264 and order by Access.

Goods will be despatched in 28 days.

Name:

Address:

.....

..... Postcode

Membership No:.....

Amount enclosed £

Oric Quickies

CASSETTE LABELS

Mr Panting from Reading has written a short program to produce box labels. If you use many makes of tapes it ensures uniformity when viewing the cassette rack.

As you will see from the examples shown, the program will produce labels for music tapes as well as computer tapes so it is quite versatile.

It has been written for the Atmos and the printer used is a Centronics one.

The listing continues on the following page.

CASSETTE No. 1

=====

Side One:	Speed: FAST
3D MAZE	
EXCELLENT GRAPHICS	

Side Two:	Speed: SLOW
BREAKOUT	
FAST ACTION GAME	

.....

3D MAZE

BREAKOUT

.....

CASSETTE No. 2A

=====

Side One:	Speed: 1.75 ips
JIM REEVES	
BEST OF..	

Side Two:	Speed: 1.75 ips
GEORGE JONES	
THE RACE IS ON	

.....

JIM AND GEORGE

.....

```

10 CLS:PRINTCHR$(17):PAPER0:INK7
20 PRINT@8,13;"DO YOU WANT INSTRUCTIO
NS":GETA$
30 IFA$="Y"THEN GOSUB1000
40 GOSUB500
50 LPRINT"+
60 LPRINT:LPRINT
70 LPRINT" CASSETTE No. ";C$
80 LPRINT"=====
=====
90 IFW$="N"THEN100ELSEGOTO110
100 LPRINT" Side One:"
110 IFW$="Y"THENLPRINT" Side One:
      Speed: ";F$
120 LPRINT" ";N$
130 IFX$="N"THEN140ELSE150
140 LPRINT:LPRINT:LPRINT
150 IFX$="Y"THENLPRINT" ";D$:LPRINT:
LPRINT
160 IFY$="N"THEN170ELSEGOTO180
170 LPRINT" Side Two:"
180 IFY$="Y"THENLPRINT" Side Two:
      Speed: ";S$
190 LPRINT" ";M$
200 IFP$="N"THEN210ELSE220
210 LPRINT:LPRINT:LPRINT:LPRINT
220 IFP$="Y"THENLPRINT" ";E$:LPRINT:
LPRINT:LPRINT
230 LPRINT"+.....
.....+"
240 IFI$="Y"THEN270
250 LPRINTCHR$(27);CHR$(14);" ";N$
260 LPRINTCHR$(27);CHR$(14);" ";M$
270 IFI$="Y"THENLPRINT:LPRINTCHR$(27)
;CHR$(14);" ";T$
275 LPRINT".....
....."
280 LPRINT:LPRINT:LPRINT:LPRINT:LPRIN
T:LPRINT
290 LPRINT"+
      +"
300 CLS:GOTO40
500 CLS:PRINT:PRINT
510 INPUT"ENTER CASSETTE No. ";C$:PRI
NT
520 INPUT"DO YOU WISH TO ENTER RECORD
ING SPEED";W$:PRINT
530 IFW$="Y"THENINPUT"ENTER SPEED SID
E 1";F$:PRINT
540 INPUT"ENTER ARTIST/PROGRAM SIDE 1
";N$:PRINT

```

CASSETTE LABELS

```

550 INPUT"DO YOU WISH TO ENTER DETAIL
S";X$:PRINT
560 IFX$="Y"THENINPUT"ENTER DETAILS S
IDE 1";D$:PRINT
570 INPUT"ENTER ARTIST/PROGRAM SIDE 2
";M$:PRINT
580 INPUT"DO YOU WISH TO ENTER RECORD
ING SPEED";Y$:PRINT
590 IFY$="Y"THENINPUT"ENTER SPEED SID
E 2";S$:PRINT
600 INPUT"DO YOU WISH TO ENTER DETAIL
S";P$:PRINT
610 IFP$="Y"THENINPUT"ENTER DETAILS S
IDE 2";E$:PRINT
620 INPUT"DO YOU WANT A SINGLE TITLE"
;I$:PRINT
630 IFI$="Y"THENINPUT"ENTER MAJOR TIT
LE";T$
640 RETURN
1000 CLS:PRINT:PRINT:PRINT:PRINT
1010 PRINT" This program enables you
to produce"
1020 PRINT"cassette box labels."
1030 PRINT" The label may be printed
on most of"
1040 PRINT"the popular printers avail
able for"
1050 PRINT"the Oric computers."
1060 PRINT"Several options are inbuil
t within"
1070 PRINT"the program, such as choic
e of speed,"
1080 PRINT"single or double titles on
the spline,"
1090 PRINT"program detail entry,etc."
1100 PRINT" Preferably,a heavier typ
e of paper"
1110 PRINT"should be used in the prin
ter,or if"
1120 PRINT"your machine will take a t
hin card"
1130 PRINT"so much the better."
1140 PRINT" After the label is print
ed,simply"
1150 PRINT"cut from '+' to '+' and fo
ld along"
1160 PRINT"the dotted lines."
1170 PRINT@10,22;CHR$(140)+"PRESS KEY
TO START":GETQ$
1180 CLS:RETURN

```

DECIMAL & BINARY CONVERTER

This useful Quickie from Spain will convert decimal to binary and vice versa.

```

10 CLS:PRINT:PRINT"Decimal or Binary?
":GETS$
20 IFS$="D"THEN100
25 IFS$="B"THEN200
30 GOTO10
100 CLS:PRINT:INPUT"What decimal numb
er";N
105 IF N>33554431 THEN 100
107 CLS:P=N:M=0
110 REPEAT:M=M+1
115 C=N/2
120 IFC>INT(C)THENB=1
125 IFC=INT(C)THENB=0
130 N=INT(N/2)
135 B$=STR$(B):B$=RIGHT$(B$,1)
140 PLOT38-M,1,B$
145 UNTILC<1
150 PRINT:PRINTSPC(25-M)P" = ":PRINT:
PRINT
155 END
200 CLS:PRINT:INPUT"What binary numbe
r";N$
205 FORU=1TOLEN(N$)
210 M$=MID$(N$,U,1):D=VAL(M$)
215 IFD<00RD>1THEN300
220 S=S+D*2^(LEN(N$)-U)
225 NEXT
230 PRINT:PRINTN$" = "S:PRINT:PRINT:E
ND
300 PRINT:PRINT"Your number isn't bin
ary":S=0:WAIT150:GOTO200

```

J. GRANADOS

Hints & Tips

<h2>SCREEN DUMP</h2>	<h2>PLAY COMMAND</h2>
<p>This short program will print the contents of the screen exactly on to the printer.</p>	<p>Here's a useful tip for users trying to get to grips with the play command.</p> <p>'D' can be 0-65535 but for this program 500-1500 is about right.</p> <p>All Wait commands are variable, depending on the user's time and patience.</p> <p>There's plenty of scope for those who wish to make improvements but hopefully this will give some readers a clearer understanding of the Play command.</p>
<pre> 5500 C\$="COPYING" 5510 FORC=1TOLEN(C\$) 5520 POKE48000+C,ASC(MID\$(C\$,C,1)):NEXT 6000 REM SCREEN COPY ROUTINE 6005 LPRINTCHR\$(17) 6010 FORX=0TO23:FORY=2TO39 6020 S=SCRN(Y,X) 6025 IF S=95 THENLPRINTCHR\$(#5B)CHR\$(#8)CHR\$(#3D);:GOTO6050 ' POUND SIGN 6026 IF S=63THEN6050 ' QUESTION MARK 6030 LPRINT CHR\$(S); 6050 NEXTY:LPRINTCHR\$(17):LPRINTCHR\$(#0B):NEXTX 6060 FORC=1TO7:POKE48000+C,32:NEXT </pre>	<pre> 20 POKE#26A,10:POKE#BBA3,0 30 CLS:PAPER4:INK7 40 FOR C=0TO7 45 N=1 50 FOR E=0TO7 60 D=500 70 PRINT@1,6;CHR\$(131) 80 PRINT@8,6;"C","N","E","D" 90 PRINT@1,8;CHR\$(131) 100 PRINT@2,8;"PLAY" 110 PRINT@6,8;CHR\$(135) 120 PRINT@7,8;C 130 PRINT@15,8;N 140 PRINT@23,8;E 150 PRINT@31,8;D 160 WAIT 300 170 NEXT E 180 WAIT 10 190 NEXT C 200 PLAY0,0,0,0 210 POKE#26A,3:POKE#BBA3,7 220 CLS:PAPER7:INK0:LIST </pre>
<h2>T.C. BROWN</h2>	
<h2>ORIC ATMOS VARIABLE LIST ROUTINE</h2>	
<p>As the ORIC ATMOS only recognizes the first two characters of a variable name, it is useful to know what has been used so far, especially on long programs. This short routine by D. Wieckowski uses no variables its self so will not add to the list it produces.</p> <p>It may be JOINED onto the end of any BASIC program (provided the line numbers are less than 63992), at any time, and will list out all the run-time variables so far used. The length of the routine is less than 250 bytes so should not give any 'out of memory error', even with a large program already in memory. The routine is run by a GOTO 63992, as run would clear all the variables in memory.</p>	<h2>ORIC ATMOS VARIABLE LIST ROUTINE</h2> <pre> 63992 DOKE(0),DEEK(#9C) 63993 PRINTCHR\$(PEEK(DEEK(0)));CHR\$(PEEK(1+DEEK(0))); 63994 IFPEEK(DEEK(0))>128THENPRINT"%"; 63995 IFPEEK(DEEK(0))<128ANDPEEK(1+DEEK(0))=>128THEN PRINT"\$"; 63996 IFDEEK(0)=>DEEK(#9E)ANDPEEK(DEEK(0))>32THENPRINT"dim";:GOTO63998 63997 PRINT:DOKE(0),(7+DEEK(0)):GOTO63999 63998 PRINT:DOKE(0),(DEEK(0)+DEEK(2+DEEK(0))) 63999 IFDEEK(0)<DEEK(#A0)THENGOTO63993ELSESTOP </pre> <h2>D. WIECKOWSKI</h2>

Features

THE ORIC IQ164

IQ164 – SPECIFICATION

The new ORIC IQ164 is a very versatile machine and has features 'built-in'. It contains 64K of RAM and up to 112K of ROM. Its operating system is cartridge based which allows many different languages to be run on the same machine. The machine's features are listed below with a brief description of each.

General

The IQ164 uses 6502 microprocessor and uses 64K of dynamic RAM and allows for the use of up to 112K of ROM. It achieves the large ROM space by paging the top 16K of addressable memory eight times. The chip count is very low because of the widespread use of ULA technology. The ULAs are used to control the following functions; video, RAM refresh, address decoding, memory paging, bus buffering and various logic functions. The case has been purpose designed and houses a high quality QWERTY keyboard. Available keyboard languages are English, French and German. All Atmos software will run on the IQ164.



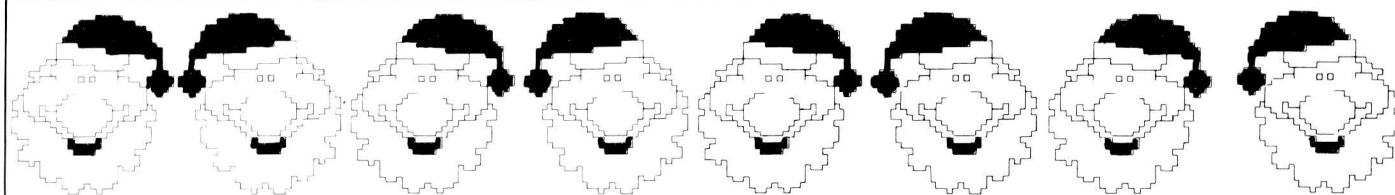
Features

Cartridges <p>The IQ164 has two cartridge slots, one for the language cartridge and one for the USER applications cartridge. The cartridges are small and are completely enclosed when they are in position. The language cartridge can contain up to 64K and the user cartridge up to 48K. The operating system supplied as standard with each machine is 'Super Extended BASIC', which contains all existing BASIC V1.1 commands, the disc operating system and a further thirty-one new commands (see software specification). Other languages which can be run include LOGO and FORTH. Applications cartridges available will include Prestel, Word Processor, Spread Sheet etc. The cartridges will run immediately after reset and are not dependent upon discs or cassette loading systems.</p>	Joystick <p>Two joystick ports are provided which will accept ATARI compatible joysticks.</p>
Video <p>The display can either be on a monitor or a UHF television receiver. The monitor output is RGB and Sync., and it can also be used with the Peritel System. The UHF output is Band 4, Channel 37, PAL or NTSC colour. The UHF output can also be B/W inverse video if requested. The video modes are as follows:-</p> <ol style="list-style-type: none"> 1. Text 28 rows × 40 columns, serial attributes. 2. Hi-Res Graphics 240 × 200 pixels, serial attributes. 3. Text 26 rows × 80 columns, serial attributes. 4. Graphics 160 × 200, 4 bits/pixel (i.e. RGBI). 5. Hi-Res Graphics 240 × 200, parallel attributes. <p>Only mode one is not bit mapped.</p>	Modem <p>A modem is available which plugs into the IQ164 which allows it to be used as a Prestel terminal and to communicate with other IQ164's via the telephone system.</p>
	Reset <p>A reset switch is provided and allows three types of reset:-</p> <ol style="list-style-type: none"> a) a full reset (with disc boot-up). b) an NMI type reset. c) a full reset, without disc boot-up.
	Software <p>The enhancements to the software over the ATMOS are probably more striking than the hardware enhancements. First of all the STRATOS is fully ATMOS compatible. A command called ATMOS initialises the STRATOS to 100% compatibility with the ATMOS. The new software features over the ATMOS are:-</p> <ol style="list-style-type: none"> 1. RENUM 2. AUTO 3. ED – a new and better screen editor. 4. IRS232 – used to define the parameter of the serial I/O channel. 5. SLIST 6. SPRINT 7. SINPUT – read serial input. 8. SETFUN – used to define the function of each key when combined with the FUNCT key. BASIC single key entry. 9. GLOAD – load game cartridge. 10. ABSDRAW 11. MOVE 3D 12. DRAW 3D 13. ADRAW 3D 14. DSET 3D 15. PAINT 16. SPLOT 17. ELLIPSE 18. EVAL – a new string handling instruction. 19. GDIR – game cartridge directory. 20. ENV – a new sound command to synthesize waveforms. 21. NOTE – new sound command. 22. ENGLISH 23. FRENCH 24. DELETE – deletes lines of program in BASIC. 25. ECLOAD – new super cassette system. 26. XSAVE 27. XLOAD 27. XSTORE – For modem communications to other ORIC's 29. SRECALL 30. ATMOS 31. ECSAVE
Disc Drives <p>The IQ164 provides a Schugart compatible disc interface. Up to four disc drives can be connected at any one time. There is sufficient capacity in the power supply to feed two ORIC slave disc drives in addition to the IQ164. ORIC disc drives use 3" discs and can store up to 160K bytes per side (see DOS commands for further details).</p>	
Cassette <p>The IQ164 has three cassette loading and saving routines which include the two ATMOS routines which run at 300 and 2400 baud. The extra routine runs at 2400 baud and saves the data in 1.5K blocks and it uses error correcting codes to give extra reliability.</p>	
RS232 <p>The RS232 uses the standard 25 pin DIN connector and can be used to send or receive data between printers, terminals, IQ164's etc. It can be set up for baud rates between 50 and 19,200 baud, 5, 6, 7 or 8 bit words, 1 or 2 stop bits and parity or no parity.</p>	
Printer <p>A parallel printer interface is provided on the rear panel.</p>	
Expansion Port <p>The address, data and control buses are available via a connector on the rear panel which allows for future peripherals.</p>	

Hints & Tips

<h2>DISASSEMBLING THE ORIC-1</h2>	<h2>FORTH</h2>
<p>These hints were discovered by Mr Turner whilst he was using his disassembler. Most of them will not work on the Atmos.</p>	<p>Whilst learning Forth from a book, Mr Harrison came across a definition for the 10 times-table which used the Forth word 'J'. This however is not available on the Oric implementation. He thus tried to resolve his problem and his first attempt is screen 1.</p> <p>However, Forth is meant to run best without using variables to the extent they are used in Basic. With a bit of trial and error using stack manipulators, he came up with the following definition of 'J' : J R> R> R> I ROT ROT >R >R SWAP >R ;</p> <p>Screen 2 incorporates this definition. Both screens illustrate the use of '.R' with prints the tables neatly in columns.</p>
<p>POKE 555,64 – disables the reset button PEEK (520) – gives a different value for each key pressed, for 56=no key 132=Space Bar 172=left arrow 156=up arrow 180=down arrow 188=right arrow DEEK (621) – location the screen starts at PEEK (623) – number of lines on screen</p> <p>The above can be poked to give crude screen windowing (Peek these values first to find out their normal values):</p> <p>DOKE621,48600:POKE623,8:CLS</p> <p>DOKE 27,27 – Oric hangs up if CTRL C used to break a program POKE48035,12 – causes the caps message to flash PEEK(781) – equals 2 if no sound is present at the din socket or 18 if a tone is present. Connect up your recorder as you are loading in a program without typing 'CLOAD' and run the following program:</p> <p>10 PRINT PEEK(781):GOTO10</p> <p>CALL 63618 – Resets paper, ink and characters CALL 63562 – Orics boot routine. This is executed on power-up CALL 52234 – clears the text screen CALL 62506 – sets Hires but Oric thinks it's in text mode. Try the following:</p> <p>CALL 62506:WAIT200:PLOT10,2,"123456"</p> <p>CALL 64199 – Zap CALL 62485 – Shoot CALL 62488 – Explode</p> <p>CALL 64196:PLAY0,0,0,0 CALL 64198:PLAY0,0,0,0 – Sound effects</p>	<p>SCREEN #1</p> <pre> Ø (TIMES-TABLE 1) 1 Ø VARIABLE OL 2 :TTI CLS 29 EMB CR 3 11 1 DO I OL ! 4 11 1 DO I OL @ * 4 .R 5 1 INK 6 PAPER 6 LOOP 7 CR LOOP 29 EMIT ; 8 2 LOAD 9 1Ø </pre>
<h2>P. TURNER</h2>	<p>SCREEN#2</p> <pre> Ø (TIMES-TABLE 2) 1 :J R> R> R> I ROT ROT >R >R SWAP >R ; 2 : TT2 CLS 29 EMIT CR 3 11 1 DO 4 11 1 DO I J * 4 .R 5 4 INK 3 PAPER 6 LOOP 7 CR LOOP 29 EMIT ; 8 ;S 9 1Ø </pre> <h2>MR. HARRISON</h2>

"Dear Oric Owner"



BYTES FREE

I have recently received the Atmos upgrade and am very pleased with it.

I note that the bytes free displayed at the start indicates approximately 10K less than the Oric-1. I assume some bytes will have been used in the new Basic, but surely not 10K. Could you please explain what happened?

J. Whittles, Rossendale

The amount of memory available for use in Basic remains exactly the same; the bytes free displayed on the Atmos takes account of the memory that would be used for the Hires screen.

PEACE OF MIND

May we enquire as to the current position on "Oric". Recently the Atmos has been disappearing from all the retail shops, as has all the software. The number of mail shots received from yourselves recently suggest the "Oric" is alive and well, as does the news that Oric are to bring out a larger computer in the new year – (noted in a recent magazine). Perhaps you could answer this point generally by comment.

D. Townsend, Essex.

It is quite obvious to most of our readers that the Oric Atmos has, over the summer months, lost its pride of place in the high street shops.

It is now apparent that Oric failed to consolidate this once favourable position by not continuing to promote brand identity through the media.

Instead, they concentrated their efforts on tapping the very lucrative overseas markets with the result that the machine achieved the acclaim and success it deserves.

However, renewed emphasis is now being given to the home market which recently has been translated into some very favourable press reports.

KEYBOARD SYMBOLS

As an absolute beginner trying to get friendly with his Oric Atmos, I would like to commend the Company for the excellent Manual provided.

However, there is one area of information, vital to the beginner, which is not adequately covered. That is, the use and function of the keyboard symbols.

What, for example is the difference in use of the different types of brackets?

K. Davies, Watford.

There are many symbols on the keyboard that are not used for programming, they are simply implemented for convenience.

The square and curly brackets are useful when outputting text, since they are used in the English language.

PROBLEM PAGE

We would like to start up a Problem Page for those of you having specific problems with programming and machine code. David Sinfeld, who I'm sure you will remember from his practical machine code articles we've printed, has offered to answer your letters, if we receive enough of them to start up this new idea.

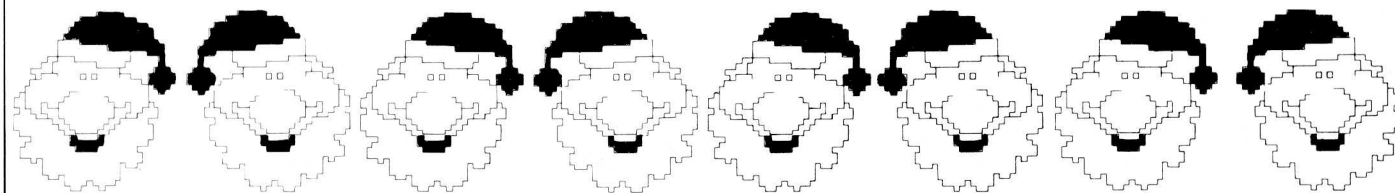
This is an offer you can't refuse!

ORIC DUAL JOYSTICK INTERFACE

I heard that Oric are releasing a dual Joystick Interface. Will it be able to work on any existing software? When will it be available and how much will it cost?

R D Waterman, St. Albans

The Joystick Interface is not programmable, and therefore any software for it will have to be specially written. Tansoft's Insect Insanity and IJK's Cybermen are compatible with it, existing titles are being adapted. For further information turn to the news page.



ORIC WINS £10000.00



After some thought, and over the next couple of days, I began to develop a loose idea for the program. It was based on the fact that I spend so much time surveying dwellings, for replacement window enquiries, and that so many enquiries never turn into definite orders. I spend a lot of time chasing potential orders, and after my own time, and the vans running costs are taken into account, have wasted a certain amount of money, if the order doesn't materialise. My program idea was to have the Oric installed in our shop, whereby customers could give details by telephone, or in person, and obtain an instant verbal or written quotation. This then would form the basis to follow up serious enquiries. The program had to be complicated enough to handle the various designs of windows, types of fitting, ages of dwelling, etc., and simple enough for the untrained shop staff to use. It had to be flexible enough to allow me to update prices, and designs, and accurate enough to avoid the quotations being too far out (in price).

Now the sparkle of Christmas had gone, all the Aliens had been zapped, and the little dots eaten in the Maze game, my Oric retired to a corner of the living room, to gather dust. It wasn't long before I began to feel frustrated with myself, at having spent some £130.00 on the new computer, just to play Space Invaders, and the like. I soon realised that there was more to the complex machine, than just games potential, and began dabbling in BASIC. After another (!) look through the Oric Manual, I knew I'd have a hard time learning serious programming from this document, and so went on the hunt for decent books about Oric. I discovered these to be few and far between, around Christmas, as the Oric hadn't exactly established itself, and publishers only produced for big selling machines (e.g. Spectrum etc.). However, I aquired copies of "The Oric 1 and how to get the most out of it" by Ian Sinclair, and "The Oric Programmer" by S M Magee & Mike James. Soon I was working away and learning the facilities Oric could offer. Incidentally, I would recommend either or both of these books to budding programmers (their contents overlap a little, but both contain valuable information, and explain things to enable beginners to understand easily).

On one of my many, regular, visits to the local computer store, I picked up a competition leaflet inviting entries for (I quote) "programs which could have commercial, scientific, industrial, or educational applications", which set me thinking. The key word I thought was "applications", and being in business myself (as a Do-it-yourself shop owner, and Building Contractor), I could think of several areas in the business, which could benefit from the application of a computer. I felt though that my lack of programming knowledge would be a hinderance, but nevertheless knew my ideas were sound enough, and decided to enter the competition.

Now Oric's graphics are excellent, and the sound is very good, and I decided to make full use of these facilities in the program. Not having any programming experience (at all!), I pondered on how to tackle the project best. I put all my ideas on paper – what information I required to be keyed into the computer, and what I required as ouptut. From here I divided the plan into blocks, these being in the order in which I thought they should be carried out by the machine. Over the next 4 or 5 weeks, at night and weekends, the program slowly took shape. Sometimes I would sit to the small hours of the morning, only going to bed when I was too tired to think straight. Using the paper plan as a guide, I sat night after night, with Oric on my knee, and the two programming books by my side. I followed the logical sequence – title, introduction, menu, routines (for 5 various operations).

Introduction:

This involved only TEXT mode, and consists of a single page brief introduction to what the program does. Using paper and INK commands, colours changed between the program screens. This allows distinction between program functions and gives a professional appearance. The press any key to continue instruction allows readers of various speeds to continue to Menu when required.

Menu:

Any of Oric's commands or functions that I wasn't sure of were referenced in one of the two programming books. The

Features

Menu is fairly straightforward TEXT display, with several IF/THEN tests afterwards to test what is keyed in, to divert the program to each of the routines (e.g. to price windows press 1). If an improper response is keyed in, then another IF/THEN test detects this, and asks the question again. While typing in the first two stages I got to like the semi-rubberised Oric keyboard. The TAB bug was often encountered and dealt with as instructed in the Manual.

HIRES display of window designs:

The first section displays white windows on a black background. The HIRES mode was called here, and extensive use of DRAW and CURSET commands made a reasonable display. Using PLOT several index letters were placed under the various window designs on screen. Oric have included a neat feature, which allows the bottom three lines of HIRES screen to be used exactly as the text screen is. This enabled me to request INPUTs and so on, using normal text commands (e.g. What is the window design A-F? or What is the window height ?). After the INPUT stage, the Menu returns, to do further work, or print the quotation.

Routines:

The routines all contain different operations and calculations. LOOPS were set up to store the information in ARRAYS, later to be recalled for printer and screen use. Three things I would urge beginners to do are, firstly, write all your variable and string names down, and secondly, do not change them during program development. Thirdly, make all variable and string names suit their subject (even if abbreviated) e.g. Window Price could be WP, or Glass Total could be GT. This will save a lot of heartbreak at debugging stage, particularly with the more subtle bugs. It could become very confusing to neglect this advice. One very useful Oric feature is the Basic sound commands. These can be used in LOOPS to provide warnings, or prompts. I used them where prompt was required, where care was necessary, and where Data could be lost by pressing the wrong key. Particular difficulty was encountered in trying to emulate the Print a command (lacking in the Oric), but TAB was used to place screen output in columns, and a special subroutine called to align decimal point (for neat layout).

Printer section:

This was the most complicated, and tedious section to program. The printer accepts commands from Basic, such as Lprint. But to make for a more flexible arrangement, it is better to use Print CHR\$(). The CHR\$ code, in brackets (), has the effect of either telling the printer what to print, or telling it to perform a special task (such as line feed etc., or turning on and off the special print faces). This sounds easy in theory, and is, but transferring these codes from the Epson Manual to the actual program is tedious. However, the time spent doing so is worthwhile, as some nice effects can be achieved in the printed copy.

End Routine:

The final routine which may be called from Menu, is the End/New Order routine. This either clears all variables and resets everything to prepare for a New Order, or goes into a bright Warning display (with bell ringing effect), to warn the user that pressing <T>erminate loses all Data in memory. At this stage a final ultimatum is given, and the user may go back to Menu, or End the run. At all stages of the program, checks for correct input (by the user) are made, and re-requested if wrong.

The program is used daily in the shop, and should save around £400-£500 per year in wasted journeys. I intend to develop it further, to cover doors and doorframes, for example, which also form a large part of my business. The recent purchase of the Oric Disc Drive should increase the usefulness of the program. I intend to convert it for Disc, which should make it quicker, and allow me to store seldom used subroutines (such as average fitting times for doors and windows), thereby extending its flexibility further.

Several persons in the trade have commented that I have a useful and marketable product and I am seriously considering either having it marketed, or transferred to a more serious business machine for marketing.

All this goes to prove that Oric could be considered for serious business use. This of course would be subject to software houses coming up with the goodies (which so far has not happened). I would urge all Oric owners (in fact all Micro owners) to have to go themselves. It's ideas that matter. Programming skills come with lots of time and practice.

I subsequently entered the Competition, and from 37 other entrants (some of whom had several thousands pounds worth of equipment), won first prize of £1000.00, and a beautiful cut crystal award. It gave me enormous satisfaction to have my work recognised by others. To think that a £130.00 computer, a £250.00 printer, and a computer novice, could win such a competition. It goes to prove that it's the software idea that matters, and not technical perfection or equipment.



In the next issue we will list the program that won Paul Jenkins his £1000.00.

Oric Owner 67

ORIC & ATMOS 48k Cassettes



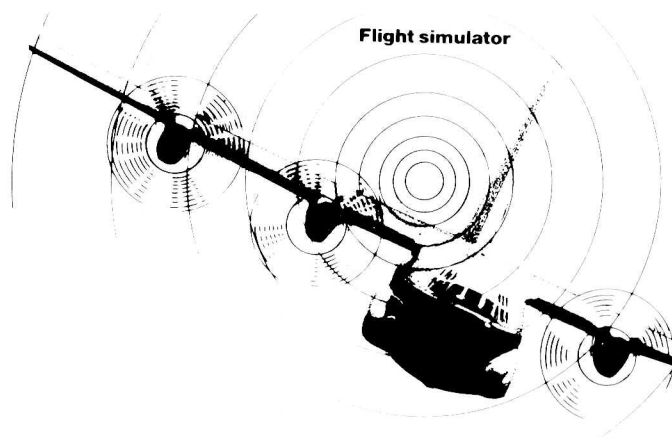
Super 3D arcade action Invaders will test your skill and 3D co-ordination to their limits. 8 rippling colour Invader screens of increasing difficulty with keyboard controlled 3D laser movement and fire sequence. (Brain required) **£6.95**

Oricle (Smart Oric)

Hours of family fun. Oric gets smarter the longer you play, eventually you will have no secrets. Be careful what you say. **£6.95** Also in French)

Mystery Tower

Superb musical arrangements. 47 different rooms. 3D and colour make this adventure game most enjoyable. You will certainly get lost, or forget where you're going. **£6.95** (Also in French)

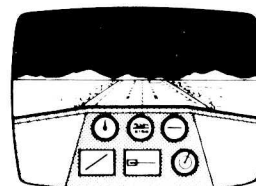


Flight simulator



Flight Simulator

With airspeed, radar, ground/vertical flight path indicators, artificial horizon and altimeter with continuous flight monitor on engine condition, stall speed etc., full colour, Hi-res graphics, 3D and sound. **£6.95**



Space Crystal

Space trading adventure game. Fascinating 3D space travel on your voyage in search of the illudium crystal, Black holes, meteorite storms, Space pirates, etc. Full colour and sound. **£6.95**. Also available for Dragon 32, Spectrum and Commodore 64

QUARK DATA P.O. BOX 61, Swindon, Wilts. Tel. (0793) 40661.

Please supply

Prices incl. p & p UK only

Cheque for total amount enclosed

Name

Address

